

# Understanding and Improving Contextualized Word Embeddings for Unsupervised Lexical Ambiguity Resolution

A dissertation submitted for the degree of  
Doctor of Philosophy (Ph.D.)

Department of Computational Linguistics,  
Heinrich Heine University Düsseldorf

by

**Deniz Ekin Yavaş**

First Supervisor: Prof. Dr. Laura Kallmeyer  
Second Supervisor: Prof. Dr. Benoît Crabbé  
Submission Date: April 2026



## Abstract

Unsupervised lexical ambiguity resolution is an important challenge in Natural Language Processing. While supervised approaches can achieve high performance, they rely on sense inventories and annotated data that are scarce and incomplete, limiting their ability to capture the full complexity of word meaning in real-world language use. Unsupervised approaches have the potential to overcome these limitations by enabling systems that automatically adapt to new domains, languages, and emerging senses.

Unsupervised approaches rely on the distributional hypothesis, grouping word instances into senses based on contextual similarity. For this reason, contextualized word embeddings of pre-trained transformer language models are particularly promising representations. Despite their success in supervised settings, however, unsupervised performance with these embeddings remains limited. Since unsupervised approaches rely primarily on the embeddings without using any external resources or annotations, understanding the properties of these embeddings and overcoming their limitations are essential for advancing unsupervised lexical ambiguity resolution.

In this thesis, we address this challenge from both theoretical and practical perspectives. We investigate how contextualized word embeddings encode and organize sense information and, based on these theoretical insights, we develop methods to improve their effectiveness for unsupervised lexical ambiguity resolution. We first investigate whether contextualized word embeddings encode sufficient information for resolving complex lexical ambiguity phenomena in supervised settings. To this end, we focus on detecting copredication with inherently polysemous nouns. We show that these embeddings indeed encode sufficient information for this task and that this information transfers cross-lingually in zero-shot settings. We further show that this task is challenging even for human experts, contextualizing the performance of computational models on this task.

Next, we diagnose a key source of limited unsupervised performance. We systematically identify which contextual features determine embedding similarity between word instances. Our results show that sense-irrelevant features — such as morphological and syntactic properties of words — have a greater effect on embedding similarity than sense-relevant features, leading to poor unsupervised performance. We address this problem by proposing an adversarial forgetting framework to remove morphological and syntactic features from contextualized word embeddings. Our results show that the resulting feature-invariant embeddings achieve better unsupervised performance than the original model embeddings.

Finally, we experiment with incorporating external sense knowledge into the embeddings through fine-tuning. We particularly focus on how topological properties — such as isotropy, uniformity, and sense-alignment — relate to task performance. Furthermore, we compare different fine-tuning methods to examine how they affect embedding space topology. Our results show that isotropy is a critical property alongside sense-alignment for creating successful sense-enhanced embeddings. We further show that different fine-tuning methods achieve different balances among topological properties, leading to differences in task performance.

Collectively, our findings reveal that what makes contextualized word embeddings effective for unsupervised lexical ambiguity resolution is more nuanced than

whether they encode sufficient information. Their effectiveness also depends on whether irrelevant information introduces noise and on how information is organized in the embedding space. Critically, the identified limitations of the embeddings can be addressed post-pretraining through the methods we propose, without the computational cost of training models from scratch. Overall, this thesis advances our understanding of contextualized word embeddings, particularly regarding the properties that determine their effectiveness in unsupervised tasks. It further contributes methods that address the identified limitations of these embeddings to improve unsupervised performance.

## Zusammenfassung

Die unüberwachte Auflösung lexikalischer Mehrdeutigkeit stellt in der Computerlinguistik eine wichtige Herausforderung in der Computerlinguistik dar. Obwohl überwachte Ansätze hohe Leistungen erzielen können, sind sie auf Bedeutungsinventare und annotierte Daten angewiesen, die nur spärlich vorhanden und unvollständig sind, was ihre Fähigkeit einschränkt, die volle Komplexität von Wortbedeutungen im realen Sprachgebrauch zu erfassen. Unüberwachte Ansätze haben das Potenzial, diese Einschränkungen zu überwinden, indem sie Systeme ermöglichen, die sich automatisch an neue Domänen, Sprachen und neu entstehende Bedeutungen anpassen.

Unüberwachte Ansätze stützen sich auf die Distributionshypothese und gruppieren Wortinstanzen anhand kontextueller Ähnlichkeit in Bedeutungsklassen. Aus diesem Grund sind kontextualisierte Worteinbettungen vortrainierter Transformer-Sprachmodelle besonders vielversprechende Repräsentationen. Trotz ihrer Erfolge in überwachten Umgebungen bleibt die unüberwachte Leistung mit diesen Einbettungen jedoch begrenzt. Da unüberwachte Ansätze primär auf den Einbettungen beruhen, ohne externe Ressourcen oder Annotationen zu verwenden, sind das Verständnis der Eigenschaften dieser Einbettungen und die Überwindung ihrer Einschränkungen wesentlich für die Weiterentwicklung der unüberwachten Auflösung lexikalischer Mehrdeutigkeit.

In dieser Dissertation begegnen wir dieser Herausforderung sowohl aus theoretischer als auch aus praktischer Perspektive. Wir untersuchen, wie kontextualisierte Worteinbettungen Bedeutungsinformationen kodieren und organisieren, und entwickeln auf Grundlage dieser theoretischen Erkenntnisse Methoden zur Verbesserung ihrer Wirksamkeit für die unüberwachte Auflösung lexikalischer Mehrdeutigkeit. Zunächst untersuchen wir, ob kontextualisierte Worteinbettungen ausreichend Informationen kodieren, um komplexe lexikalische Mehrdeutigkeitssphänomene in überwachten Umgebungen aufzulösen. Zu diesem Zweck konzentrieren wir uns auf die Erkennung von Koprädikation mit inhärent polysemen Nomen. Wir zeigen, dass diese Einbettungen tatsächlich ausreichend Informationen für diese Aufgabe kodieren und dass diese Informationen in Zero-Shot-Szenarien von einer Sprache zur anderen übertragbar sind. Darüber hinaus zeigen wir, dass diese Aufgabe selbst für menschliche Experten anspruchsvoll ist, was die Leistung computergestützter Modelle in einen geeigneten Kontext setzt.

Anschließend diagnostizieren wir eine wesentliche Ursache für die begrenzte Leistungsfähigkeit unüberwachter Methoden. Wir identifizieren systematisch, welche kontextuellen Merkmale die Einbettungsähnlichkeit zwischen Wortinstanzen bestimmen. Unsere Ergebnisse zeigen, dass bedeutungsirrelevante Merkmale — wie morphologische und syntaktische Eigenschaften von Wörtern — einen stärkeren Einfluss auf die Einbettungsähnlichkeit haben als bedeutungsrelevante Merkmale, was zu einer schlechten unüberwachten Leistung führt. Wir begegnen diesem Problem durch die Entwicklung eines adversarialen Vergessensframeworks, um morphologische und syntaktische Merkmale aus kontextualisierten Worteinbettungen zu entfernen. Unsere Ergebnisse zeigen, dass die resultierenden merkmalsinvarianten Einbettungen eine bessere unüberwachte Leistung erzielen als die ursprünglichen Modelleinbettungen.

Abschließend untersuchen wir die Integration externen Bedeutungswissens in die Einbettungen durch Feinabstimmung. Wir konzentrieren uns dabei insbesondere darauf, wie topologische Eigenschaften — wie Isotropie, Uniformität und Bedeutungsalignierung — mit der Aufgabenleistung zusammenhängen. Darüber hinaus vergleichen wir verschiedene Feinabstimmungsmethoden, um zu untersuchen, wie sie die Topologie des Einbettungsraums beeinflussen. Unsere Ergebnisse zeigen, dass Isotropie neben der Bedeutungsalignierung eine entscheidende Eigenschaft für die erfolgreiche Verbesserung von Einbettungen hinsichtlich lexikalischer Bedeutungen ist. Wir zeigen ferner, dass verschiedene Feinabstimmungsmethoden unterschiedliche Gleichgewichte zwischen topologischen Eigenschaften erreichen, was zu Leistungsunterschieden führt.

Insgesamt zeigen unsere Befunde, dass das, was kontextualisierte Worteinbettungen für die unüberwachte Auflösung lexikalischer Mehrdeutigkeit wirksam macht, vielschichtiger ist als die Frage, ob sie ausreichend Informationen kodieren. Ihre Wirksamkeit hängt auch davon ab, ob irrelevante Informationen Rauschen einführen und wie die Informationen im Einbettungsraum organisiert sind. Entscheidend ist, dass die identifizierten Einschränkungen der Einbettungen nach dem Vortraining durch die von uns vorgeschlagenen Methoden behoben werden können, ohne den Rechenaufwand des Trainings von Grund auf. Insgesamt erweitert diese Dissertation unser Verständnis kontextualisierter Worteinbettungen, insbesondere hinsichtlich der Eigenschaften, die ihre Wirksamkeit für unüberwachte Aufgaben bestimmen. Darüber hinaus leistet sie einen methodischen Beitrag zur Behebung der identifizierten Einschränkungen dieser Einbettungen, um die unüberwachte Leistung zu verbessern.

# Contents

|                                                                                                                                  |            |
|----------------------------------------------------------------------------------------------------------------------------------|------------|
| <b>Abstract</b>                                                                                                                  | <b>i</b>   |
| <b>Zusammenfassung</b>                                                                                                           | <b>iii</b> |
| <b>List of Figures</b>                                                                                                           | <b>vii</b> |
| <b>Abbreviations</b>                                                                                                             | <b>ix</b>  |
| <b>1 Introduction</b>                                                                                                            | <b>1</b>   |
| 1.1 Motivation . . . . .                                                                                                         | 1          |
| 1.2 Theoretical Background and Related Work . . . . .                                                                            | 3          |
| 1.2.1 Lexical Ambiguity . . . . .                                                                                                | 3          |
| 1.2.2 Pre-Trained Transformer Language Models . . . . .                                                                          | 7          |
| 1.2.3 Computational Approaches to Lexical Ambiguity Resolution . . . . .                                                         | 13         |
| 1.2.4 Lexical Ambiguity Resolution with Contextualized Word Em-<br>beddings . . . . .                                            | 19         |
| 1.3 Problem Statement . . . . .                                                                                                  | 22         |
| 1.3.1 Effects of Context . . . . .                                                                                               | 22         |
| 1.3.2 Creating Invariant Embeddings . . . . .                                                                                    | 25         |
| 1.3.3 Creating Sense-Enhanced Embeddings . . . . .                                                                               | 26         |
| 1.4 Research Goals, Questions, and Hypotheses . . . . .                                                                          | 27         |
| 1.5 Summary of the Research Papers . . . . .                                                                                     | 29         |
| 1.5.1 Cross-lingual Copredication Detection (Chapter 2) . . . . .                                                                | 30         |
| 1.5.2 Copredication Corpus Annotation (Chapter 3) . . . . .                                                                      | 32         |
| 1.5.3 Feature Relevance for WSD (Chapter 4) . . . . .                                                                            | 33         |
| 1.5.4 Feature Removal with Adversarial Forgetting (Chapter 5) . . . . .                                                          | 35         |
| 1.5.5 Sense Knowledge Enhancement through Fine-Tuning (Chap-<br>ter 6) . . . . .                                                 | 36         |
| 1.6 Scope . . . . .                                                                                                              | 38         |
| 1.7 Thesis Overview . . . . .                                                                                                    | 40         |
| <b>2 Identifying Semantic Argument Types in Predication and Copred-<br/>ication Contexts: A Zero-Shot Cross-Lingual Approach</b> | <b>41</b>  |
| <b>3 Cococorpus: a corpus of copredication</b>                                                                                   | <b>53</b>  |
| <b>4 Assessing the Significance of Encoded Information in Contextual-<br/>ized Representations to Word Sense Disambiguation</b>  | <b>65</b>  |

|          |                                                                                                                       |            |
|----------|-----------------------------------------------------------------------------------------------------------------------|------------|
| <b>5</b> | <b>Improving Word Sense Induction through Adversarial Forgetting of Morphosyntactic Information</b>                   | <b>79</b>  |
| <b>6</b> | <b>On the Relation Between Fine-Tuning, Topological Properties, and Task Performance in Sense-Enhanced Embeddings</b> | <b>95</b>  |
| <b>7</b> | <b>Discussion and Conclusion</b>                                                                                      | <b>113</b> |
| 7.1      | Summary and Discussion of the Findings . . . . .                                                                      | 113        |
| 7.1.1    | Copredication as Complex Ambiguity Phenomenon (RQ1) . .                                                               | 114        |
| 7.1.2    | Contextual Features and Sense-Relevance (RQ2) . . . . .                                                               | 116        |
| 7.1.3    | Removing Sense-Irrelevant Features (RQ3) . . . . .                                                                    | 119        |
| 7.1.4    | Creating Sense-Enhanced Embeddings through Fine-Tuning (RQ4) . . . . .                                                | 121        |
| 7.1.5    | Discussion and Synthesis . . . . .                                                                                    | 122        |
| 7.2      | Contributions . . . . .                                                                                               | 123        |
| 7.2.1    | Theoretical Contributions . . . . .                                                                                   | 123        |
| 7.2.2    | Practical Contributions . . . . .                                                                                     | 124        |
| 7.2.3    | Resource Contributions . . . . .                                                                                      | 125        |
| 7.3      | Limitations . . . . .                                                                                                 | 126        |
| 7.4      | Future Research Directions . . . . .                                                                                  | 127        |
| 7.5      | Concluding Remarks . . . . .                                                                                          | 127        |
|          | <b>Bibliography</b>                                                                                                   | <b>129</b> |

# List of Figures

|      |                                                                                                                                      |     |
|------|--------------------------------------------------------------------------------------------------------------------------------------|-----|
| 1.1  | Adversarial forgetting approach. . . . .                                                                                             | 10  |
| 1.2  | BERT embedding clusters. . . . .                                                                                                     | 12  |
| 1.3  | BERT embeddings before and after isotropy post-processing. . . . .                                                                   | 13  |
| 1.4  | Static and contextualized word embeddings. . . . .                                                                                   | 18  |
| 1.5  | Different computational approaches for lexical ambiguity resolution. .                                                               | 19  |
| 1.6  | Influence of sense-relevant and sense-irrelevant features on embedding<br>similarity. . . . .                                        | 23  |
| 1.7  | Embedding spaces with different uniformity. . . . .                                                                                  | 26  |
| 1.8  | Predication classification method. . . . .                                                                                           | 30  |
| 1.9  | Copredication detection method. . . . .                                                                                              | 31  |
| 1.10 | Cluster explainability method. . . . .                                                                                               | 33  |
| 1.11 | Framework for training a forget-gate to create feature-invariant em-<br>beddings. . . . .                                            | 35  |
| 1.12 | SCL and SPL fine-tuning methods for sense knowledge enhancement.                                                                     | 37  |
| 7.1  | Copredication detection method. . . . .                                                                                              | 115 |
| 7.2  | PCA visualization of the embeddings of ‘ask’ in different sense sen-<br>tences and important features found in each cluster. . . . . | 118 |
| 7.3  | PCA visualizations of BERT embeddings and morphology-invariant<br>embeddings of ‘area’ instances. . . . .                            | 120 |



# Abbreviations

**CL** Contrastive Learning

**KNN** K-Nearest Neighbors

**NLP** Natural Language Processing

**PLM** Pre-trained Transformer Language Model

**SCL** Supervised Contrastive Learning

**SPL** Supervised Predictive Learning

**T-PAS** Typed Predicate Argument Structures

**UCL** Unsupervised Contrastive Learning

**WiC** Word-in-Context

**WSD** Word Sense Disambiguation

**WSI** Word Sense Induction



# Chapter 1

## Introduction

### 1.1 Motivation

Lexical ambiguity occurs when a single word form has multiple possible senses (Cruse, 1986). This phenomenon is widespread in natural language: most words have more than one sense, and the number of senses tends to increase with word frequency (Zipf, 1945). For instance, the noun ‘bank’ can denote a financial institution (1-a) or riverbank (1-b); the noun ‘head’ can denote a body part (2-a), leadership position (2-b), or top part of an object (2-c).

- (1) bank:
  - a. I deposited money at the **bank**. SENSE: *financial institution*
  - b. I fished from the **bank**. SENSE: *riverbank*
- (2) head:
  - a. I hit my **head** on the doorframe. SENSE: *body part*
  - b. I was appointed as **head** of the department. SENSE: *leadership*
  - c. The **head** of the hammer hit the floor. SENSE: *top part*

Linguistic context, the surrounding words or sentences, provides the primary signal for resolving such ambiguities. This is closely linked to the *distributional hypothesis* (Harris, 1954): words appearing in similar contexts tend to have similar meanings. Consequently, a specific sense tends to co-occur with similar neighboring words (Navigli, 2012). As a result, to distinguish different senses of the word ‘bank’ in (1), we can use the surrounding words as cues. In (1-a), words like ‘deposited’ and ‘money’ signal the financial institution sense of ‘bank’, whereas in (1-b), the verb ‘fished’ signals the riverbank sense.

Accurately capturing sense distinctions is essential for successful Natural Language Processing (NLP) systems for both language understanding and generation tasks. Machine translation systems must determine whether ‘bank’ refers to a financial institution or riverbank to select the appropriate translation. Similarly, information extraction systems must distinguish whether ‘cell’ denotes a biological cell, a prison cell, or a mobile phone to correctly classify entities. Given that context provides the primary signal for distinguishing senses, successful NLP systems must effectively leverage contextual information.

Indeed, in recent years, Pre-trained Transformer Language Models (PLMs) like BERT (Devlin et al., 2019) and GPT (Radford et al., 2018) have become highly

successful and popular due to their ability to capture context efficiently and over long ranges. These models produce *contextualized word embeddings* that vary with surrounding text, capturing rich contextual information specific to each instance. This makes them well-suited for capturing sense distinctions and resolving lexical ambiguity. For instance, contextualized word embeddings for ‘bank’ differ between the contexts in (1-a) and (1-b) because the word co-occurs with different words in each context. In principle, this should enable these embeddings to represent different senses distinctly.

However, in practice, these embeddings show inconsistent performance across different settings for lexical ambiguity resolution. While supervised methods can successfully leverage them (Vial et al., 2019; Blevins and Zettlemoyer, 2020; Nair et al., 2020), unsupervised methods show limited performance (Mosolova et al., 2025). In supervised methods, contextualized word embeddings serve as input features to classifiers that are trained on sense-annotated data. These methods successfully learn decision boundaries that separate different senses of words in the embedding space. In contrast, unsupervised methods must discover sense distinctions without annotated examples, typically by analyzing patterns of embedding similarity — measuring how close embeddings are in the embedding space. These methods typically assume that the embeddings of the same sense instances are closer together in the embedding space than other sense instances, leading them to exhibit higher embedding similarity.

The success of supervised methods demonstrates that contextualized word embeddings contain the necessary information for distinguishing senses. Yet the poor performance with unsupervised methods reveals an interesting puzzle: if these embeddings successfully encode meaning in context, why do similarity-based methods fail to exploit this information for ambiguity resolution? This gap challenges a widespread assumption in NLP, which underlies many applications: that contextualized word embeddings approximate word meaning in context, with embedding similarity reflecting semantic similarity.

Several factors may explain this gap. First, contextualized word embeddings encode not only *sense-relevant* information, such as surrounding content words, but also *sense-irrelevant* linguistic and surface-level information, such as certain morphological properties, casing, or position of the words (Dalvi et al., 2022; Sajjad et al., 2022). As a result, higher embedding similarity might reflect shared sense-irrelevant features rather than shared sense. Second, beyond what information is encoded, how this information is organized in the embedding space can also affect task performance. In previous studies, two topological properties — *isotropy* (Mu and Viswanath, 2018) and *uniformity* (Wang and Isola, 2020) — have been linked to the expressiveness of contextualized word embeddings (Gao et al., 2021; Rajaei and Pilehvar, 2021; Zhang et al., 2022) and to performance on semantics-related tasks (Biš et al., 2021; Gao et al., 2021; Liu et al., 2021a,b; Rajaei and Pilehvar, 2021; Yan et al., 2021). Topological properties of the embeddings may also determine how effectively sense-relevant information in the embeddings can be exploited.

These complexities raise several critical questions about the role of embedding properties in unsupervised lexical ambiguity resolution: How well do these embeddings align with senses? Which contextual features determine embedding similarity between word instances, and are they sense-relevant? Does the topology of the embedding space limit how effectively sense-relevant information can be exploited?

Addressing these questions has both theoretical and practical implications. Theoretically, understanding how contextualized word embeddings represent and organize sense information can help us understand what determines their effectiveness in lexical ambiguity resolution tasks. Practically, these insights are essential for developing methods to overcome their limitations and better leverage their potential. This has direct implications for building robust NLP applications. Supervised methods, despite their success, cannot scale to the full complexity of human language. Extensive sense inventories exist for only a small number of languages and specialized domains. Furthermore, sense inventories have inherent limitations: they cover only a limited set of words, fail to capture all possible senses, and become easily outdated as language evolves.<sup>1</sup> Successful unsupervised ambiguity resolution, in contrast, could resolve ambiguity without relying on these resources, enabling systems to automatically adapt to new domains, languages, and senses as they emerge in real-world language use.

Building on these motivations, this thesis explores these questions through five published research papers. Together, they investigate how contextualized word embeddings encode and organize sense information, and, based on these insights, develop methods to improve their effectiveness for unsupervised lexical ambiguity resolution. The rest of the Introduction chapter is organized as follows. [Section 1.2](#) provides the theoretical background and related work. [Section 1.3](#) formulates the problem statement. [Section 1.4](#) presents the research goals, questions and hypotheses. [Section 1.5](#) presents the summary of the research papers that comprise this thesis. [Section 1.6](#) defines the scope of the investigation. Finally, [Section 1.7](#) provides a structural overview of this thesis.

## 1.2 Theoretical Background and Related Work

This section provides the theoretical background and related work relevant to this thesis. It is organized into four parts. [Section 1.2.1](#) provides an overview of lexical ambiguity types relevant to this thesis and the linguistic properties that distinguish them. [Section 1.2.2](#) examines Pre-trained Transformer Language Models (PLMs) in detail, covering their architecture, applications, and the properties of the contextualized word embeddings they produce. [Section 1.2.3](#) provides an overview of the computational approaches and resources relevant to lexical ambiguity resolution, including tasks, sense inventories, input representations, and supervised and unsupervised approaches. Finally, [Section 1.2.4](#) reviews how contextualized word embeddings of PLMs represent senses and how they are used and improved for lexical ambiguity resolution.

### 1.2.1 Lexical Ambiguity

Lexical ambiguity arises when a single word form is associated with more than one meaning ([Cruse, 1986](#)). In this section, we provide an overview of lexical ambiguity types relevant to this thesis and the linguistic properties that distinguish them.

---

<sup>1</sup>New senses emerge frequently in language, e.g., ‘cloud’ in computing, ‘viral’ as social media success.

### 1.2.1.1 Homonymy and Polysemy

Lexical ambiguity can occur differently in language, with the clearest distinction being between homonymy and polysemy. This distinction is motivated by the degree of semantic relatedness between senses (Klepousniotou, 2002; Klepousniotou and Baum, 2007; Klepousniotou et al., 2008, 2012). Homonymy occurs when the senses of a word are semantically unrelated, while polysemy occurs when senses share semantic connections. The examples with ‘bank’ and ‘head’ in (1) and (2) illustrate this contrast. The *financial institution* and *riverbank* senses of ‘bank’ are not semantically related to each other; in contrast, different senses of ‘head’ can be seen as extensions from a single core meaning.

However, a single word form can exhibit both homonymous and polysemous relations across different sense pairs. For instance, while the two senses of ‘bank’ shown in (3) are homonymous (*financial institution* vs. *riverbank*), the *financial institution* sense itself can exhibit polysemous relations to other senses, as shown in (4). The *location* and *organization* senses in (4) are semantically related to the *financial institution* sense of ‘bank’.

- (3) bank (homonymy):
- a. I deposited money at the **bank**. SENSE: *financial institution*
- b. I fished from the **bank**. SENSE: *riverbank*
- (4) bank<sub>financial institution</sub> (polysemy):
- a. The **bank** in the center is closed for renovation. SENSE: *location*
- b. The **bank** approved my loan application. SENSE: *organization*

Homonymy and polysemy can be further divided into subtypes. Homonymy can also occur across part-of-speech categories, as illustrated in (5).

- (5) duck:  
 a. The **duck** swam across the pond. PART-OF-SPEECH: *noun*  
 SENSE: *waterfowl*  
 b. Please **duck** your head. PART-OF-SPEECH: *verb*  
 SENSE: *to lower*

This thesis does not investigate this type of homonymy, as these cases can be readily resolved using syntactic and morphological cues rather than semantic ones. For instance, the syntactic context in (5-a) and (5-b) directly determines the part-of-speech of ‘duck’ and in turn its sense, making semantic cues unnecessary for ambiguity resolution.

### 1.2.1.2 Subtypes of Polysemy

Polysemy can be further divided into subtypes that exhibit more complex patterns. The primary distinction among polysemy subtypes is between *regular* (or *systematic*) and *irregular* (or *non-systematic*) *polysemy* (Apresjan, 1974). Regular polysemy follows systematic patterns of meaning shift that apply to multiple lexical items in a language (Apresjan, 1974) and often extends cross-lingually (Dölling, 2020).

For instance, the ANIMAL-MEAT pattern is one such pattern and has many instances in English: ‘chicken’, ‘lamb’, ‘turkey’, and ‘fish’ can all refer to both the

living animal and the meat derived from it. Furthermore, this pattern appears in many languages (Dölling, 2020). In contrast, irregular polysemy involves meaning shifts specific to individual words, with no systematic relation between senses. For instance, ‘wire’ can refer to both a metal strand and a listening device. These two senses share meaning similarities, which makes them polysemous rather than homonymous (Brocher et al., 2016). However, this meaning relation is specific to this word and does not extend systematically to other lexical items.

The literature identifies numerous underlying patterns of regular polysemy (see Dölling, 2020, for an overview), including:

- CONTAINER-FOR-CONTENTS (*container* vs. *content*): bottle, glass, pot
- ANIMAL-MEAT (*animal* vs. *food*): chicken, lamb, turkey
- PLANT-FRUIT (*tree* vs. *fruit*): apple, cherry, orange

### 1.2.1.3 Inherent Polysemy and Copredication

A particularly important subtype of regular polysemy is *inherent* (or *logical*) *polysemy* (Pustejovsky, 1998; Asher, 2011; Pustejovsky and Jezek, 2012), where nouns systematically have multiple, complementary senses as part of their core meaning (Dölling, 2020). Senses of inherently polysemous nouns are also referred to as *meaning facets* (Chen et al., 2022, 2025a). The defining characteristic distinguishing inherent polysemy from other types of regular polysemy is that no single sense (or *meaning facet*) can be identified as prior or derivative and all senses are equally fundamental to the meaning of the word (Dölling, 2020).

For instance, ‘book’ has two meaning facets simultaneously: an INFORMATION facet, referring to the information content (as in “I understood the book.”), and a PHYSICAL\_OBJECT facet, referring to the artifact that contains the information (as in “I lifted the book.”), with neither facet being more fundamental than the other for the word’s meaning. Moreover, both facets together constitute the meaning of the word. This is in contrast to other types of regular polysemy, where one sense historically derives from another. For instance, the *animal* sense of ‘chicken’ can be considered prior to the *meat* sense (Dölling, 2020).

Since inherent polysemy is a subtype of regular polysemy, it also exhibits systematic cross-lingual patterns. Each pattern includes multiple nouns, and these patterns often extend cross-lingually (Pustejovsky and Jezek, 2012). The following examples illustrate some of these patterns:

- PHYSICAL\_OBJECT•INFORMATION: book, newspaper, journal, report
- EVENT•FOOD: lunch, dinner, breakfast, brunch
- ORGANIZATION•LOCATION: university, school, hospital, bank

**Predicate Relations and Copredication.** Each facet of an inherently polysemous noun corresponds to a distinct semantic type (e.g., INFORMATION, PHYSICAL\_OBJECT, EVENT, LOCATION). It is widely adopted in the literature to refer to these nouns using dot notation, combining both semantic types. Therefore, they are also referred to as *dot objects*, *dot types* or *complex type nouns* (Pustejovsky, 1995,

1998; Pustejovsky and Jezek, 2012). For instance, ‘book’ has two semantic types corresponding to each of its facets: PHYSICAL\_OBJECT and INFORMATION, and it is a PHYSICAL\_OBJECT•INFORMATION complex type noun. In contrast, nouns that do not exhibit such polysemy are referred to as *simple type nouns*, possessing only one meaning facet that corresponds to one semantic type. For instance, ‘rumor’ is only INFORMATION type, while ‘rock’ is only PHYSICAL\_OBJECT type.

Predicates require arguments with certain semantic types, as defined by their argument structure (Pustejovsky, 1995). Because inherently polysemous nouns have more than one semantic type, they interact with predicates differently than simple type nouns. Predications involving simple type nouns primarily involve direct type matching (Pustejovsky and Jezek, 2012).<sup>2</sup> The predicates in (6) and (7) require arguments with specific semantic types: ‘understand’ requires an INFORMATION, while ‘lift’ requires a PHYSICAL\_OBJECT type argument. When these predicates are combined with different semantic type arguments, the result is a semantically anomalous sentence.

- (6) rumor (INFORMATION):
  - a. I *understood* the **rumor**.
  - b. #I *lifted* the **rumor**.
- (7) rock (PHYSICAL\_OBJECT):
  - a. #I *understood* the **rock**.
  - b. I *lifted* the **rock**.

In contrast, inherently polysemous nouns like ‘book’, which has multiple semantic types simultaneously, can satisfy the selectional requirements of predicates requiring either facet. For instance, ‘book’ can be combined with a predicate that requires an INFORMATION type argument, as in (8-a), or that requires a PHYSICAL\_OBJECT type argument, as in (8-b), without causing semantic anomaly. Furthermore, these nouns can also combine with predicates that simultaneously select both of their facets, and this can happen with a single predicate that requires the complex type itself as an argument (Pustejovsky, 1998). For instance, predicates like ‘read’ require the complex type PHYSICAL\_OBJECT•INFORMATION as an argument because reading involves both the physical object and processing its informational content, as in (8-c).

- (8) book (PHYSICAL\_OBJECT•INFORMATION):
  - a. I *understood* the **book**. FACET: INFORMATION
  - b. I *lifted* the **book**. FACET: PHYSICAL\_OBJECT
  - c. I *read* the **book** on marine biology. FACET: PHYSICAL\_OBJECT•INFORMATION

More interestingly, this simultaneous selection can occur through multiple predicates that require different semantic type arguments. This phenomenon is known as *copredication* (Pustejovsky, 1995, 1998; Asher, 2011; Pustejovsky and Jezek, 2012). For instance, in (9-a), both facets of the inherently polysemous noun (‘book’) are

---

<sup>2</sup>One exception to this is *coercion* (Pustejovsky and Jezek, 2012), where a predicate forces a type shift in its argument. For instance, in “finish the soup”, the verb ‘finish’ (which requires an EVENT argument) coerces ‘soup’ from PHYSICAL\_OBJECT type to an EVENT interpretation (the event of eating or cooking the soup is finished).

selected by different predicates in the same context: ‘understand’ selects INFORMATION, while ‘lift’ selects PHYSICAL\_OBJECT. Applying similar coordination to simple type nouns results in semantic anomaly, as shown in (9-b). Similarly, with other types of regular polysemy, the same coordination typically produces a zeugmatic reading (Cruse, 1986; Dölling, 2020), as shown in (9-c).

- (9) a. **Inherent Polysemy:**  
I *understood* and *lifted* the **book**.  
b. **Simple Type Noun:**  
#I *understood* and *lifted* the **rock**.  
c. **Regular Polysemy:**  
#John *lifted* and *drank* the **cup**.

Example (9-b) is semantically anomalous because ‘rock’ lacks an INFORMATION facet. Example (9-c) is zeugmatic because the CONTAINER and CONTENT senses of ‘cup’ cannot be simultaneously active. This ability to license copredication is what distinguishes inherent polysemy from other subtypes of polysemy (Dölling, 2020).

## 1.2.2 Pre-Trained Transformer Language Models

PLMs have achieved remarkable success across diverse Natural Language Processing (NLP) tasks, leading to widespread adoption in both research and applications. This section explores these models in detail, covering their architecture, applications, and the properties of the contextualized word embeddings they produce. [Section 1.2.2.1](#) covers their underlying architecture. [Section 1.2.2.2](#) details encoder-based models. [Section 1.2.2.3](#) discusses how pre-trained encoder language models are used for specific tasks, and [Section 1.2.2.4](#) discusses approaches for improving these models. Finally, [Section 1.2.2.5](#) takes a closer look at the contextualized word embeddings they produce, reviewing the literature that focuses on what information they encode and how the information is organized in the embedding space.

### 1.2.2.1 Transformer Architecture

The transformer architecture (Vaswani et al., 2017) became the foundation for modern contextualized language models. The key innovation of this architecture is the self-attention mechanism. Self-attention allows the model to attend to other tokens in the sequence during the prediction of a token. This allows the model to encode rich contextual information both locally and globally and enables more effective parallel processing and modeling of long-range dependencies. These advantages have contributed to the superior performance of transformer language models over earlier contextualized models such as ELMo (Peters et al., 2018), which relied on the LSTM architecture (Hochreiter and Schmidhuber, 1997).

PLMs can be broadly categorized into two main architectural variants. *Encoder models* are typically designed to build rich contextual representations of language. The most widely used encoder models belong to the BERT family, including BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019b), and ALBERT (Lan et al., 2019). They employ *bidirectional self-attention* that allows the model to attend to both preceding and following context. These models are pre-trained with a *masked language modeling* objective, as illustrated in (10-a), where they predict masked tokens

using context from both directions. This pre-training creates powerful contextual representations that support a wide range of downstream language understanding tasks such as sentiment analysis, named entity recognition, and Word Sense Disambiguation (WSD).

*Decoder models* are typically designed for language generation. The most widely used decoder models belong to the GPT family, including GPT-1 (Radford et al., 2018), GPT-2 (Radford et al., 2019) and GPT-3 (Brown et al., 2020). They employ *unidirectional (causal) self-attention* that allows each token to attend only to preceding tokens in the sequence, preventing access to future context. These models are pre-trained with a *causal language modeling* objective, as illustrated in (10-b), where the model predicts the next token based solely on the preceding context. This architecture makes decoders particularly effective for language generation tasks such as paraphrasing, story generation, and text summarization.

- (10)    a.    **Masked Language Modeling (Bidirectional):**  
              I ate an [MASK] after lunch.  
          b.    **Causal Language Modeling (Unidirectional):**  
              I ate an

### 1.2.2.2 Pre-Trained Encoder Language Models: BERT and Its Variants

**Architecture.** BERT consists of multiple transformer layers, each containing a multi-head self-attention mechanism and a feed-forward network. BERT is available in two configurations: BERT-base (12 layers, 768 hidden size, 12 attention heads, 110M parameters) and BERT-large (24 layers, 1024 hidden size, 16 attention heads, 340M parameters).

**Training.** BERT is pre-trained on two objectives. The first objective is *masked language modeling*, where 15% of tokens are masked and predicted from bidirectional context. The second objective is *next sentence prediction*, where the model predicts whether two sentences are consecutive in the data. This helps the model learn sentence-level relations. The model is pre-trained on large text corpora: BookCorpus (800M words) (Zhu et al., 2015) and English Wikipedia (2500M words).

**Variants.** BERT’s success led to numerous variants and extensions. Monolingual variants include RoBERTa and ALBERT for English, which improve on the original BERT model, and language-specific models such as those for Italian (Staatsbibliothek and Schweter, 2025) and German (Chan et al., 2020, GBERT). Finally, multilingual variants are trained on multilingual corpora, creating shared representations across multiple languages, such as mBERT (Devlin et al., 2019) and XLM-RoBERTa (Conneau et al., 2020).

### 1.2.2.3 Use of Pre-trained Encoder Language Models

Pre-trained encoder models can be used for specific tasks through several approaches, including fine-tuning, feature extraction, and cross-lingual transfer:

**Fine-tuning.** The most widely used approach with encoder models is fine-tuning, where the PLM undergoes additional training on a specific task. Task-specific output

layers are added on top of the model and the model parameters are updated to adapt to the target task. Fine-tuning allows leveraging the knowledge gained during pre-training and adapts it to the target task. The most common fine-tuning strategy is Supervised Predictive Learning (SPL), where the model is fine-tuned by minimizing the Cross-Entropy loss to predict class labels.

**Feature Extraction.** PLMs can serve as feature extractors, generating contextualized word embeddings to be used by downstream models. In this approach, embeddings can be extracted from different layers of the PLM, e.g., the final layer, a specific intermediate layer, or averaged across multiple layers. These embeddings then serve as input to task-specific models or algorithms, such as classifiers or clustering algorithms. The key difference from fine-tuning is that model parameters remain frozen during feature extraction, rather than being updated to adapt to the task.

**Cross-Lingual Transfer with Multilingual Models.** Multilingual PLMs enable *cross-lingual transfer*, where a model trained on one language (*source language*) can be applied to other languages (*target language*) with minimal or no additional training. This is typically achieved either by fine-tuning the multilingual model on a downstream task, or by training task-specific models using the multilingual embeddings as input (using the multilingual PLM as a feature extractor). This can be done in a *few-shot* setting, where a small amount of target language data is used for training, or a *zero-shot* setting, where no target language data is used at all. Previous studies have shown that multilingual models successfully perform zero-shot cross-lingual transfer across tasks such as named entity recognition, part-of-speech tagging, and question answering (Pires et al., 2019; Conneau et al., 2020; Wu and Dredze, 2020; Aghazadeh et al., 2022), performing well on multilingual benchmarks such as XTREME (Hu et al., 2020) and XGLUE (Liang et al., 2020). While monolingual models can achieve high single-language performance, cross-lingual transfer with multilingual PLMs is useful in low-resource settings.

#### 1.2.2.4 Improving Pre-trained Encoder Language Models

PLMs can be improved through various approaches to adapt them to specific domains or tasks, or to improve their overall performance. Key approaches adopted in this thesis include:

**Knowledge Enhancement.** Knowledge enhancement methods incorporate structured knowledge into model representations to improve performance on tasks that can benefit from external knowledge that may not be gained from standard language modeling objectives alone, such as sense knowledge. This can be achieved during the pre-training phase of the model or also by fine-tuning (for a comprehensive overview, see Hu et al., 2023).

**Invariant Representation Learning.** *Invariant representation learning* aims to create representations that are invariant to certain features learned from the data — such as nuisances, biases, or task-irrelevant properties — while preserving task-relevant information. Creating representations invariant to such information aims

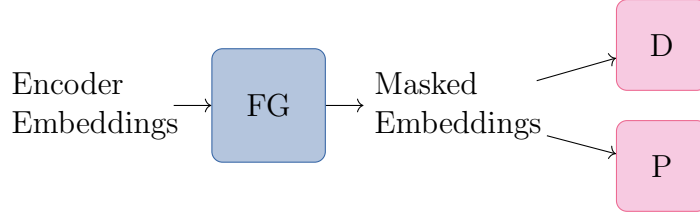


Figure 1.1: Adversarial forgetting approach. The forget-gate ( $FG$ ) is trained as part of an adversarial model alongside a discriminator ( $D$ ) and predictor ( $P$ ). The forget-gate masks the encoder’s embeddings to remove the unwanted features and is trained by minimizing the predictor’s loss while maximizing the discriminator’s loss.

to increase model performance and robustness. Examples include creating representations invariant to facial expressions in face recognition (Bronstein et al., 2003) or creating domain-invariant representations in NLP (Louizos et al., 2017; Jaiswal et al., 2018; Peng and Zhang, 2020; Xin et al., 2022).

One method used for this purpose is *adversarial forgetting* (Jaiswal et al., 2020). A forget-gate is trained to mask the embeddings of an encoder to remove unwanted features. The forget-gate is trained as part of an adversarial model with two additional components, a discriminator and a predictor, as illustrated in Figure 1.1. The encoder’s embeddings are passed to the forget-gate to be masked. Then, these masked embeddings are passed to the discriminator and predictor. The discriminator is trained to predict the unwanted features from the masked embeddings, while the predictor is trained to perform the main task, ensuring that task-relevant information is retained in the masked embeddings. The forget-gate is trained by minimizing the predictor’s loss while maximizing the discriminator’s loss. As a result, the output of the trained forget-gate is masked embeddings that are invariant to unwanted features.

**Contrastive Learning.** *Contrastive Learning (CL)* is a representation learning approach that aims to structure the embedding space to position related instances (*positives*) closer together and unrelated instances (*negatives*) further apart. CL has been used with PLMs for improving their embeddings (Gao et al., 2021; Liu et al., 2021a,b; Yan et al., 2021), task adaptation (Gunel et al., 2020) and knowledge enhancement (Mosolova et al., 2024). Both supervised and unsupervised approaches can be adopted for CL.

*Unsupervised Contrastive Learning (UCL)* requires no labeled data, and positive and negative pairs are determined without annotation. One example approach is *random span masking* (Liu et al., 2021a,b). In this approach, multiple augmented versions of each sentence are created by randomly masking its tokens. These augmented versions are treated as positives for the original sentence, referred to as the *anchor*, while all other instances in the dataset are treated as negatives. An example is shown in (11). The goal is to structure the embedding space such that each anchor is closer to its augmented versions (positives) than to any other instance (negatives).

(11) **Unsupervised Contrastive Learning:**

- **Anchor:** “The bank approved my loan.”

- **Positives (with data augmentation via random span masking):**  
 “[MASK] bank approved my loan.”, “The bank approved [MASK] loan.”, ...
- **Negatives:** All other instances

*Supervised Contrastive Learning (SCL)* uses label information to define positives and negatives of an anchor, where instances sharing the same class label are treated as positives and instances with different labels as negatives (Khosla et al., 2020). For instance, in a lexical ambiguity task, positives could be instances of the same sense and negatives could be instances of different senses, as shown in (12). The goal is to structure the embedding space such that instances of the same class are closer together and instances of different class are further apart.

(12) **Supervised Contrastive Learning:**

- **Anchor (Sense: *bank*<sub>1</sub>):** “The bank approved my loan.”
- **Positives (same sense instances (*bank*<sub>1</sub>)):**  
 “I deposited money at the bank.”, “My bank account is blocked.”
- **Negatives (different sense instances (*bank*<sub>2</sub>)):**  
 “The bank was muddy.”, “We fished at the river bank.”, ...

#### 1.2.2.5 Understanding Contextualized Word Embeddings

Understanding contextualized word embeddings produced by PLMs requires examining both what information they encode and how that information is organized in the embedding space, as both aspects are central to their effectiveness in downstream tasks.

**What Information Do Contextualized Word Embeddings Encode?** Interpretability studies aim to understand the inner workings of models. This area has become increasingly important with the emergence of PLMs, as these models function as black boxes whose internal computations are neither transparent nor interpretable. One central focus of interpretability studies is understanding what types of linguistic information PLM embeddings capture as a result of their pre-training.

Most studies approach this question by focusing on specific linguistic properties, such as syntactic, semantic, or morphological information (Tenney et al., 2019; Yaghoobzadeh et al., 2019; Zhao et al., 2020; Chersoni et al., 2021). A common method for these investigations is *probing*, which involves training (mostly linear) classifiers on frozen model embeddings to predict specific linguistic properties (Belingov et al., 2017; Conneau et al., 2018). The performance of these classifier suggests whether the targeted information is encoded in the embeddings. For instance, to investigate whether embeddings encode part-of-speech information, a probing classifier is trained to predict part-of-speech categories of the words from their embeddings. High classification accuracy indicates that this information is encoded in the embeddings.

Beyond probing for specific linguistic properties, other studies have adopted more general approaches to uncover the information encoded in these embeddings. Clustering-based methods, for instance, analyze the overall structure of the embedding space without targeting predefined linguistic categories (Dalvi et al., 2022;



Figure 1.2: Examples of BERT embedding clusters: a cluster of superlative adjectives (left) and an uninterpretable cluster (right). Figures from Dalvi et al. (2022) (left: Figure 2b, p. 5; right: Figure 5d, p. 8).

Sajjad et al., 2022). These studies apply clustering algorithms to contextualized word embeddings and examine what concepts the resulting clusters correspond to, revealing what information is encoded in the embeddings. Previous studies have shown that contextualized word embeddings simultaneously encode multiple types of information, including syntactic, semantic, and morphological information (Dalvi et al., 2022; Sajjad et al., 2022). However, these approaches also reveal that embedding clusters do not always correspond to traditional linguistic categories (Dalvi et al., 2022). For instance, Dalvi et al. (2022) have shown that while some clusters of BERT embeddings align with traditional linguistic categories such as morphological properties (*superlative adjectives*), others are uninterpretable, as shown in Figure 1.2.

Furthermore, the information encoded in contextualized word embeddings varies across model layers. Tokens are first represented as static, uncontextualized input embeddings, which do not yet incorporate any contextual information. As tokens pass through transformer layers *contextualization* occurs: the embeddings become more context-specific (Ethayarajh, 2019; Sevastjanova et al., 2021) and information from surrounding context is progressively incorporated into the input embeddings. This effect has been measured using embedding similarity in previous studies (Ethayarajh, 2019; Sevastjanova et al., 2021), showing that the embeddings of the same word instances become more dissimilar across layers as contextualization increases. Furthermore, previous studies have shown that different types of information dominate different layers of the PLMs. There is a gradual linguistic progression from morphological and syntactic information in earlier layers, to more high-level semantic information in later layers (Tenney et al., 2019). Overall, higher layers are more specialized in semantics-related tasks (Liu et al., 2019a; Tenney et al., 2019; Zhao et al., 2020).

**How is the Information Organized in the Embedding Space?** Beyond the type of linguistic information encoded in the embeddings, the topology of the embedding space can also affect the performance on semantics-related tasks (Biś et al., 2021; Gao et al., 2021; Liu et al., 2021a,b; Rajaei and Pilehvar, 2021; Yan et al., 2021). In particular, *isotropy* (Mu and Viswanath, 2018) and *uniformity* (Wang and Isola, 2020) have been identified as key properties that affect embedding *expressiveness*: their ability to capture and distinguish nuanced relations from the data (Gao et al., 2021; Rajaei and Pilehvar, 2021; Zhang et al., 2022).

*Isotropy* refers to the distribution of the embeddings across the directions of the

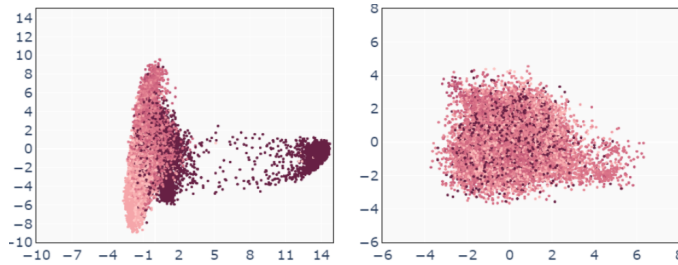


Figure 1.3: BERT embeddings before (left) and after (right) post-processing for isotropy improvement. After post-processing, embeddings are more evenly distributed across directions. Figure from [Rajae and Pilehvar \(2021\)](#) (Figures 5d and 5e, p. 584.)

embedding space. More isotropic embeddings are more evenly distributed across all directions, whereas less isotropic embeddings are concentrated in certain directions, with some directions dominating over others. *Uniformity* refers to the density of the distribution of the embeddings in the embedding space, capturing whether they are concentrated in certain areas or spread evenly.

Contextualized word embeddings have inherently limited expressiveness, as they are not evenly distributed across all directions but instead concentrated within a narrow cone in the embedding space ([Gao et al., 2019](#)), a phenomenon corresponding to low isotropy. As a result, even the embeddings of random words often exhibit high similarity ([Ethayarajh, 2019](#)). Importantly, improving the isotropy and uniformity of the embeddings of PLMs has been shown to result in better semantic task performance ([Biš et al., 2021](#); [Gao et al., 2021](#); [Liu et al., 2021a,b](#); [Rajae and Pilehvar, 2021](#); [Yan et al., 2021](#)).

Multiple approaches have been shown to improve isotropy, including post-processing methods ([Bihani and Rayz, 2021](#); [Biš et al., 2021](#); [Rajae and Pilehvar, 2021](#)) and fine-tuning with CL ([Gao et al., 2021](#); [Liu et al., 2021a,b](#); [Yan et al., 2021](#)). Similarly, CL has also been shown to improve uniformity ([Wang and Isola, 2020](#); [Gao et al., 2021](#)), leading to improved semantic task performance ([Gao et al., 2021](#)). The effect of post-processing the BERT embeddings for isotropy improvement is illustrated in [Figure 1.3](#). The resulting embeddings become more evenly distributed in terms of directions. The choice of the fine-tuning method also critically affects the embedding space topology. It has been shown that fine-tuning with SPL reduces isotropy, degrading embedding expressiveness ([Zhang et al., 2022](#)). In contrast, fine-tuning with SCL improves both isotropy and uniformity ([Gao et al., 2021](#)).

### 1.2.3 Computational Approaches to Lexical Ambiguity Resolution

The ambiguous nature of language poses challenges for NLP, as systems must determine the intended sense of a word in context. This section provides an overview of the computational approaches, tasks, and resources relevant to lexical ambiguity resolution. [Section 1.2.3.1](#) introduces the most widely used tasks for resolving lexical ambiguity computationally. [Section 1.2.3.2](#) discusses the sense inventories and resources widely used for these tasks. [Section 1.2.3.3](#) covers the word and context representations used as input features for ambiguity resolution systems. Finally,

Section 1.2.3.4 outlines the supervised and unsupervised approaches used to address these tasks.

### 1.2.3.1 Lexical Ambiguity Resolution Tasks

WSD and Word Sense Induction (WSI) are two prominent tasks for resolving lexical ambiguity computationally (Navigli, 2009, 2012). These tasks approach the problem from different perspectives. WSD involves selecting the correct sense label for a word instance in context from a predefined set of senses (Navigli, 2009), as illustrated in (13). In contrast, WSI aims to automatically induce the senses of word instances from context without relying on a pre-existing sense inventory by grouping same sense instances (Navigli, 2012), as illustrated in (14).

(13) **Word Sense Disambiguation:**

•**Input:**

–**Sentence:** “The **bank** approved my loan.”

–**Sense List:**  $\{bank_1, bank_2, bank_3, \dots\}$

•**Output:**  $bank_1$

(14) **Word Sense Induction:**

•**Input:**

–**Sentence<sub>1</sub>:** “I deposited money at the **bank**.”

–**Sentence<sub>2</sub>:** “I sat by the river **bank**.”

–**Sentence<sub>3</sub>:** “The **bank** offers savings accounts.”

•**Output:**  $\{sentence_1, sentence_3\}, \{sentence_2\}$

Standard WSD benchmarks include SENSEVAL and SemEval evaluation datasets (Edmonds and Cotton, 2001; Snyder and Palmer, 2004; Pradhan et al., 2007; Navigli et al., 2013; Moro and Navigli, 2015; Raganato et al., 2017), which are combined to form a unified benchmark in Raganato et al. (2017). For WSI, evaluation datasets include dedicated SemEval shared tasks (Agirre and Soroa, 2007; Manandhar et al., 2010; Jurgens and Klapaftis, 2013; Navigli and Vannella, 2013) and datasets derived from SemCor (Miller et al., 1993), a corpus annotated with WordNet (Fellbaum, 1998) senses, as in Mosolova et al. (2025).

With the development of PLMs, new tasks and benchmarks have emerged to evaluate these models’ capabilities. One such benchmark is SuperGLUE (Wang et al., 2019), which includes the Word-in-Context (WiC) task (Pilehvar and Camacho-Collados, 2019) relevant to lexical ambiguity resolution. This task has become widely adopted for evaluating systems that utilize PLMs. The task is formulated as a binary classification problem. Given two instances of the same word, the task is to determine whether the word has the same meaning in both contexts, as illustrated in (15).

(15) **Word-in-Context** (example from Pilehvar and Camacho-Collados, 2019):

•**Input:**

–**Sentence<sub>1</sub>:** “There’s a lot of trash on the **bed** of the river.”

–**Sentence<sub>2</sub>:** “I keep a glass of water next to my **bed** when I sleep.”

| Adjective                      | Adverb  | Noun                                                                                                                                                                                                                                 |                                                                                                                                                                              | Verb                                                                                                                                                                                                                                                          |
|--------------------------------|---------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| adj.all<br>adj.pert<br>adj.ppl | adv.all | noun.Tops<br>noun.act<br>noun.animal<br>noun.artifact<br>noun.attribute<br>noun.body<br>noun.cognition<br>noun.communication<br>noun.event<br>noun.feeling<br>noun.food<br>noun.group<br>noun.location<br>noun.motive<br>noun.object | noun.person<br>noun.phenomenon<br>noun.plant<br>noun.possession<br>noun.process<br>noun.quantity<br>noun.relation<br>noun.shape<br>noun.state<br>noun.substance<br>noun.time | verb.body<br>verb.change<br>verb.cognition<br>verb.communication<br>verb.competition<br>verb.consumption<br>verb.contact<br>verb.creation<br>verb.emotion<br>verb.motion<br>verb.perception<br>verb.possession<br>verb.social<br>verb.stative<br>verb.weather |

Table 1.1: WordNet Supersenses.

•**Output:** *False*

In recent years, different variations of these tasks have been introduced through the SemEval workshop series, such as multilingual and cross-lingual WiC (Martelli et al., 2021) and visual WSD (Raganato et al., 2023).

### 1.2.3.2 Sense Inventories and Resources

WordNet (Fellbaum, 1998) has been the most widely used sense inventory for lexical ambiguity research. WordNet is a large lexical database of English that organizes nouns, verbs, adjectives, and adverbs into sets of cognitive synonyms (*synsets*), each expressing a distinct concept. Different senses of words correspond to different *synsets*. For instance, ‘bank’ has multiple synsets corresponding to its different senses, and some of its synsets and their definitions are shown in (16).

#### (16) WordNet synsets for ‘bank’:

- bank.n.01*: sloping land (especially the slope beside a body of water)
- depository\_financial\_institution.n.01*: a financial institution that accepts deposits and channels the money into lending activities
- bank.n.03*: a long ridge or pile
- bank.n.09*: an arrangement of similar objects in a row or in tiers
- ...

Synsets are linked to each other through lexical relations, forming a network. One such relation is *hypernymy/hyponymy*, which organizes synsets into hierarchies based on is-a relations. As (17) illustrates, the synset *depository\_financial\_institution.n.01* (*financial institution* sense of ‘bank’) is a hyponym of *financial\_institution.n.01*, which is in turn a hyponym of *institution.n.01*, continuing up the hierarchy.

#### (17) WordNet hierarchy for *depository\_financial\_institution.n.01*:

- Hierarchy:**  $\rightarrow$  *financial\_institution.n.01*  $\rightarrow$  *institution.n.01*  $\rightarrow$  *organization.n.01*  $\rightarrow$  *social\_group.n.01*  $\rightarrow$  *group.n.01*  $\rightarrow$  *abstraction.n.06*

→ *entity.n.01*

•**Supersense:** *noun.group*

This hierarchy continues to more general concepts at the top level, where WordNet organizes synsets into broad semantic categories called *supersenses*. In the case of *financial institution* sense of ‘bank’, the connected supersense is *noun.group*, as shown in (17). The list of supersenses found in WordNet, with their definitions and examples, can be seen in Table 1.1.

A resource related to WordNet is SemCor (Miller et al., 1993). SemCor is a corpus of English text in which words have been annotated with their WordNet senses and supersenses. SemCor consists of over 200,000 annotated word tokens from the Brown Corpus. An example sentence from SemCor can be seen in (18).

(18) A bright sun and brisk wind had the track...  
           bright.a.01 sun.n.01           bracing.s.01 wind.n.01 get.v.03           path.n.04

Most WSD datasets use WordNet as their sense inventory (Edmonds and Cotton, 2001; Snyder and Palmer, 2004; Pradhan et al., 2007; Navigli et al., 2013; Moro and Navigli, 2015; Raganato et al., 2017). Similarly, the WiC task uses WordNet as its core resource. Finally, gold standards for some WSI tasks rely on WordNet sense distinctions (Jurgens and Klapaftis, 2013; Mosolova et al., 2025). SemCor is widely used as the primary training resource for supervised WSD systems (Agirre et al., 2010; Zhong and Ng, 2010; Raganato et al., 2017; Vial et al., 2019; Blevins and Zettlemoyer, 2020).

**Resources for Inherent Polysemy and Copredication.** There is a significant lack of datasets focusing on inherent polysemy. While WordNet is a comprehensive resource, it does not systematically provide sense distinctions for inherent polysemy. For instance, WordNet treats the INFORMATION and PHYSICAL\_OBJECT facets of *book* as a single sense, as shown in (19).

(19) book:  
       *book.n.01*: a written work or composition that has been published (printed on pages bound together). E.g., “I am reading a good book on economics.”

Similarly, although copredication has received considerable attention in theoretical linguistics, there is a significant lack of empirical studies and computational resources for investigating this phenomenon.

Since inherent polysemy and copredication involve semantic argument types, one approach is to utilize resources that provide this information. One extensive resource for this purpose is Typed Predicate Argument Structures (T-PAS) (Jezek et al., 2014), a resource of argument structures for Italian verbs. T-PAS provides patterns of argument structures for verbs based on semantic types, along with corpus examples for each pattern. For instance, (20) shows two T-PAS patterns for ‘finire’ (‘finish’), both requiring a HUMAN type subject and an EVENT type argument, with the latter realized either as a direct object in (20-a) or as a ‘di’ infinitive clause in (20-b).

(20) finire (‘finish’):

- a. **[[human]-subj] finire [[event]-obj]:**  
“Finisce l’allenamento.” (‘He finishes the training session.’)
- b. **[[human]-subj] finire [[event]-di INF [V]]:**  
“Non faccio in tempo a finire di bere la mia birra.” (‘I don’t have time to finish drinking my beer.’)

Although T-PAS was not designed specifically for copredication, it can serve as a valuable resource for this purpose, as it provides information about the semantic types that each predicate selects in a given context.

### 1.2.3.3 Word and Context Representations for Lexical Ambiguity Resolution

The choice of input features for NLP systems is critical and has evolved significantly over time. In the context of lexical ambiguity resolution, these features correspond to representations of words and their surrounding context. This is motivated by the distributional hypothesis (Harris, 1954). Instances of the same sense tend to co-occur with similar words (Navigli, 2012), and contextual similarity reflects semantic similarity (Miller and Charles, 1991). Therefore, successfully representing contextual information is crucial for lexical ambiguity resolution.

Early computational approaches mostly relied on manually selected features to capture the context of word instances, such as neighboring words, part-of-speech tags of the neighboring words, syntactic relations in the sentence, and co-occurrence statistics (Navigli, 2009). For instance, the meaning facets or senses of the PHYSICAL\_OBJECT•INFORMATION noun ‘book’ can be easily identified using the information about the predicates in sentences: ‘understand’ in (21-a) signals the INFORMATION facet, while ‘lift’ in (21-b) signals the PHYSICAL\_OBJECT facet.

- (21) book (PHYSICAL\_OBJECT•INFORMATION):
- a. I *understood* the **book**. FACET: INFORMATION  
FEATURE: ‘understood’
  - b. I *lifted* the **book**. FACET: PHYSICAL\_OBJECT  
FEATURE: ‘lifted’

Similarly, for ‘bank’ (senses: *financial institution* or *riverbank*), the information about the surrounding content words can easily resolve the ambiguity. Specifically, the words ‘deposited’ and ‘money’ in (22-a) signal the *financial institution* sense, whereas ‘fished’ in (22-b) signals the *riverbank* sense.

- (22) bank:
- a. I deposited money at the **bank**. SENSE: *financial institution*  
FEATURES: ‘deposited’, ‘money’
  - b. I fished from the **bank**. SENSE: *riverbank*  
FEATURES: ‘fished’

With the development of word embedding models such as Word2Vec (Mikolov et al., 2013) and GloVe (Pennington et al., 2014), word meaning and semantic relations between words could be learned automatically from large corpora and words could be represented with distributional representations. This eliminated the need for manually selected features for representing words in many applications.

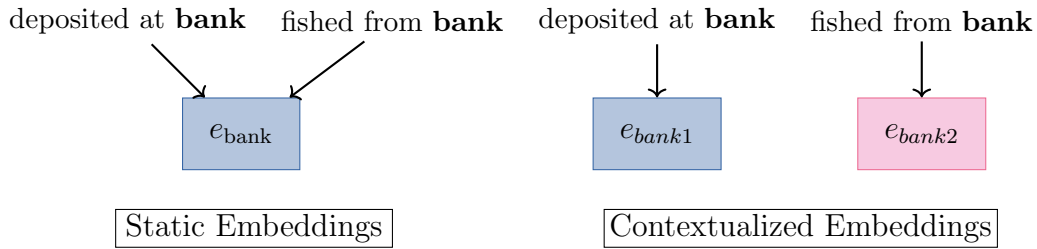


Figure 1.4: Comparison of static and contextualized word embeddings for the instances of ‘bank’.

Even though the embeddings produced by these models are powerful features for many NLP systems, they are not suitable for lexical ambiguity resolution because these models assign a single fixed representation to each word type regardless of context. They are therefore referred to as *static word embeddings*.

Contextualized word embeddings from PLMs address this limitation by generating distinct representations for each word instance based on its surrounding context. As they encode rich contextual information, these embeddings are particularly well-suited for lexical ambiguity resolution. For instance, ‘bank’ receives the same static word embedding whether it appears in the *financial institution* sense, as in (22-a), or the *riverbank* sense, as in (22-b). However, a PLM like BERT produces distinct embeddings for each instance, potentially reflecting the different senses signaled by their respective contexts. This comparison is illustrated in Figure 1.4.

In recent years, PLMs and their contextualized word embeddings have become the dominant input features for lexical ambiguity resolution. How these models are used for this purpose is discussed in detail in Section 1.2.4.

#### 1.2.3.4 Supervised and Unsupervised Approaches

Computational approaches to lexical ambiguity resolution can be broadly categorized into supervised and unsupervised methods, which differ fundamentally in their reliance on labeled data for training. Supervised approaches utilize annotated data with predefined sets of senses to train models, whereas unsupervised approaches discover sense distinctions in raw data without relying on annotation. A key limitation of supervised approaches is their reliance on large amounts of sense-annotated data, which is expensive and time-consuming to produce and not always readily available for all words, domains, or languages. This motivates the use of unsupervised methods that can be applied to raw data alone.

In the context of lexical ambiguity resolution, supervised approaches typically train classifiers to determine the correct sense label of a given word instance from a predefined set of senses (Navigli, 2012). This can be accomplished through several methods: fine-tuning PLMs, or training classifiers such as support vector machines or neural networks. The performance of supervised systems is evaluated by comparing classification results to gold standard annotations using metrics such as F-score (e.g., Raganato et al., 2017).

Unsupervised approaches rely on the distributional hypothesis, grouping word instances into sense groups based on their contextual similarity (Navigli, 2012). Clustering is a widely used method for this purpose. The resulting clusters are then evaluated against a gold standard using metrics such as adjusted rand index (e.g.,

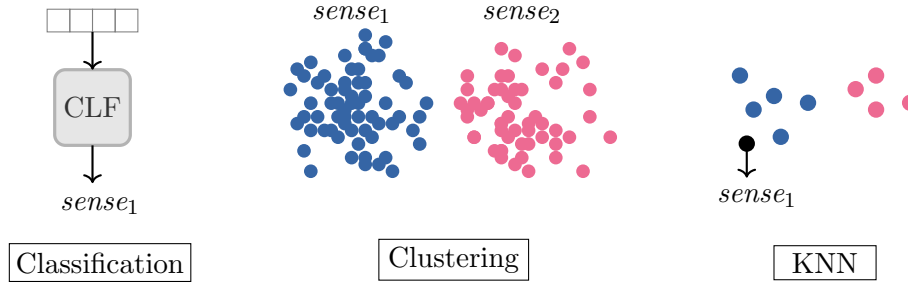


Figure 1.5: Different computational approaches for lexical ambiguity resolution: supervised classification (left), unsupervised clustering (center), and KNN classification (right).

Periti et al., 2024), purity (e.g., Periti et al., 2024), F-B<sup>3</sup> (e.g., Liétard et al., 2024), and V-measure (e.g., Komninos and Manandhar, 2016). Alternatively, embedding similarity can be used to determine whether two word instances share the same sense by measuring the similarity between their embeddings (e.g., Pilehvar and Camacho-Collados, 2019).

Some supervised methods, such as K-Nearest Neighbors (KNN), depend on the same similarity assumption as unsupervised methods. Rather than learning explicit decision boundaries, KNN classifies instances based on embedding similarity (Loureiro et al., 2021). The instances are classified based on the labels of their  $k$ -closest neighbors. Figure 1.5 provides a comparative visualization of these three approaches.

For WSD, both supervised and unsupervised methods can be used. WSI is inherently unsupervised, as it aims to automatically induce sense distinctions directly from raw data. In practice, unsupervised WSD and WSI employ similar methods, both primarily relying on contextual similarity to group word instances into sense groups. For WiC, both unsupervised and supervised approaches can be adopted: using similarity-based methods for unsupervised settings, or using classification methods for supervised settings (Pilehvar and Camacho-Collados, 2019).

### 1.2.4 Lexical Ambiguity Resolution with Contextualized Word Embeddings

PLMs have become the foundation for lexical ambiguity resolution in recent years. Systems that utilize these models show strong performance across different tasks and surpass previous state-of-the-art approaches. This section reviews how contextualized word embeddings of PLMs represent senses and how they are used and improved for lexical ambiguity resolution. Section 1.2.4.1 examines how well these embeddings naturally align with senses and how they represent different types of lexical ambiguity. Section 1.2.4.2 covers task performance across different task and approaches. Finally, Section 1.2.4.3 discusses related work focusing on improving these embeddings for lexical ambiguity resolution with sense knowledge enhancement.

#### 1.2.4.1 Alignment of Contextualized Word Embeddings with Senses

Studies have shown that contextualized word embeddings do not naturally align with senses (Yenicelek et al., 2020). Yenicelek et al. (2020) have examined the alignment between BERT embedding space and WordNet supersenses through clustering, revealing low alignment reflected in poor clustering results. Ethayarajh (2019) has further shown that embeddings vary more due to contextual variation than due to meaning variation. For instance, function words, despite not being polysemous, exhibit great variation in their embeddings because they occur in a wider variety of contexts. Garcia (2021) has demonstrated that contextual similarity can dominate semantic similarity with contextualized word embeddings. When a word, its synonym, and an unrelated word occur in the same sentential context, the word’s embedding exhibits no greater similarity to the synonym than to the unrelated word. Interestingly, Garcia (2021) has shown that static word embeddings are better at capturing this type of semantic relation, as they are not influenced by sentential context. Furthermore, previous studies have observed that embedding similarity is influenced by other factors beyond semantics-relevant information, such as syntactic properties (Yenicelek et al., 2020; Gessler and Schneider, 2021) and orthographic form (Laicher et al., 2021).

**Representation of Different Lexical Ambiguity Types.** Several studies have focused on how homonymy and polysemy are represented by PLMs. Studies have shown that homonymy and polysemy exhibit different distributions of embedding similarity (Nair et al., 2020; Haber and Poesio, 2021). Several studies have compared human sense similarity judgements to BERT embedding similarity scores, with contradictory results: some have found a degree of correlation (Haber and Poesio, 2020b,a; Nair et al., 2020; Haber and Poesio, 2021) while others have found none (Haber and Poesio, 2020b,a).

Finally, Sevastjanova et al. (2021) have shown that the degree of contextualization in BERT varies with lexical ambiguity type. As previously discussed, contextualization refers to the degree to which embeddings of the same word become dissimilar across layers (see Section 1.2.2.5 for more details). Homonymous words have less semantically related senses and therefore occur in more diverse contexts. This diversity is reflected in their embeddings, making them more strongly affected by contextualization. In contrast, polysemous words show varying degrees of contextualization depending on their degree of polysemy. Highly polysemous words exhibit stronger contextualization, whereas less polysemous words show lower contextualization due to the high semantic relatedness between senses and their occurrence in more similar contexts.

#### 1.2.4.2 Task Performance

There are several ways to leverage PLMs for lexical ambiguity resolution tasks. One approach is to *fine-tune* the model directly on the downstream task, which has shown strong performance on WSD (Huang et al., 2019; Peters et al., 2019; Blevins and Zettlemoyer, 2020) and WiC (Peters et al., 2019; Wang et al., 2019). Alternatively, the models can be used as *feature extractors*, where contextualized word embeddings serve as input features for task-specific models and algorithms. As this thesis focuses on the properties of contextualized word embeddings, the

remainder of this section reviews studies that use them as feature extractors without any task-specific adaptation.

The feature extraction approach has been effective for supervised WSD (Vial et al., 2019; Blevins and Zettlemoyer, 2020; Nair et al., 2020). For instance, even training a linear classifier using BERT embeddings as input features already achieves good results (Blevins and Zettlemoyer, 2020; Nair et al., 2020). KNN is another example of the feature extraction approach for supervised WSD. Loureiro and Jorge (2019) have shown that the performance of KNN with BERT embeddings surpasses previous WSD systems. Similarly, Wiedemann et al. (2019) have shown that KNN with BERT embeddings achieves strong performance; however, the method’s success degrades with sparse data, as sense-irrelevant features may begin to determine embedding similarity between instances. In unsupervised settings, performance drops considerably; clustering BERT embeddings for WSI fails to surpass basic heuristic baselines (Mosolova et al., 2025). However, more structured clustering approaches, such as two-step methods that combine WSI with Concept Induction through local and global clustering, have shown more promising results (Liétard et al., 2024). Finally, for WiC, both supervised and unsupervised approaches remain significantly below human performance (Pilehvar and Camacho-Collados, 2019).

**Performance Based on Lexical Ambiguity Type.** Lexical ambiguity resolution performance with PLMs and their embeddings varies across different lexical ambiguity types. Studies have shown that performance with homonymous senses is higher than polysemous senses in both supervised (Nair et al., 2020) and unsupervised settings (Haber and Poesio, 2021). Haber and Poesio (2021) have further shown that within polysemy, task performance varies across polysemy pattern types, with patterns such as FOOD•EVENT and CONTAINER-FOR-CONTENTS yielding different levels of performance.

#### 1.2.4.3 Improving Contextualized Word Embeddings with Sense Knowledge Enhancement

Several approaches have been proposed to improve PLMs embeddings for lexical ambiguity resolution. This section focuses on *sense knowledge enhancement*, i.e., incorporating external sense knowledge into contextualized word embeddings, as this is the most relevant line of work for this thesis. Some studies have incorporated external sense knowledge during pre-training of the PLMs by modifying the pre-training objectives to explicitly encode sense distinctions (Peters et al., 2019; Levine et al., 2020). These approaches have shown improvements on WSD and WiC tasks (Peters et al., 2019; Levine et al., 2020). Another approach that has been used is to incorporate sense knowledge after pre-training through fine-tuning (Mosolova et al., 2024) or post-processing methods (Bihani and Rayz, 2021). Mosolova et al. (2024) have fine-tuned BERT with SCL for this purpose and reported improved WiC performance. In addition, UCL has been shown to improve the embeddings of PLMs for lexical ambiguity resolution tasks. Liu et al. (2021b) have fine-tuned BERT and RoBERTa with UCL to create better word-in-context representations and reported improved performance on WiC and WSD.

## 1.3 Problem Statement

The previous section provided theoretical background and related work on lexical ambiguity, Pre-trained Transformer Language Models (PLMs) and their embeddings, computational approaches to lexical ambiguity resolution, and lexical ambiguity resolution with contextualized word embeddings. Building on this foundation, this section presents the problem statement, identifying three interconnected research gaps that motivate this thesis. [Section 1.3.1](#) discusses the relation between context and lexical ambiguity resolution, specifically how certain contextual information can harm unsupervised performance, and how performance with different ambiguity types can vary depending on their relation to context. [Section 1.3.2](#) builds on this discussion and motivates the development of a method for creating contextualized word embeddings invariant to harmful contextual information, i.e., sense-irrelevant features. Finally, [Section 1.3.3](#) discusses whether sense-enhanced embeddings can be created through fine-tuning and the potential trade-offs involved in terms of topological properties of the embeddings.

### 1.3.1 Effects of Context

Linguistic context is one of the primary signals for resolving lexical ambiguity, making contextual representations essential for this task. Contextualized word embeddings of PLMs have become the dominant approach for lexical ambiguity resolution tasks ([Huang et al., 2019](#); [Peters et al., 2019](#); [Blevins and Zettlemoyer, 2020](#); [Loureiro et al., 2021](#)). Unlike static word embeddings that assign a single representation per word type, these embeddings generate distinct representations for each word instance based on surrounding context, making them well-suited for capturing the contextual differences required for ambiguity resolution. However, their performance in unsupervised settings reveals limitations that require systematic investigation.

In supervised approaches, contextualized word embeddings serve as input features to classifiers that learn to identify sense boundaries using contextual information, guided by sense annotations. This approach has proven effective for lexical ambiguity resolution ([Vial et al., 2019](#); [Blevins and Zettlemoyer, 2020](#); [Nair et al., 2020](#)). Unsupervised approaches, by contrast, rely on a different assumption: that embedding similarity reflects sense distinctions. These methods assume that instances with the same sense will have similar embeddings while those with different senses will have dissimilar embeddings, and that clustering or similarity metrics can uncover these sense distinctions. However, this assumption is problematic: previous studies have shown that embeddings do not naturally align with senses ([Yenicelek et al., 2020](#)) and, indeed, unsupervised methods show limited performance ([Mosolova et al., 2025](#)).

One potential explanation for this problem is that contextualized word embeddings encode not only sense-relevant information but also sense-irrelevant information, and the latter may determine embedding similarity. This can be illustrated with the following examples. In both [\(23-a\)](#) and [\(23-b\)](#), ‘bank’ denotes different senses but has the same syntactic role (*object*). Even though the surrounding words (‘withdraw’, ‘money’ vs. ‘picnic’, ‘river’) clearly signal different senses, the similarity between ‘bank’ embeddings might be determined by their shared syntactic role (sense-irrelevant features) rather than the sense-signaling surrounding words (sense-



textual similarities as *contextual overlap*. Homonymous senses have low contextual overlap due to their diverse contexts, and polysemous senses have high contextual overlap due to their semantically similar contexts.

To illustrate this difference, we can return to the examples (24) and (25) from earlier. We can expect different senses of ‘bank’ to occur in more diverse contexts than different senses of ‘head’. The senses of ‘bank’ (*financial institution* vs. *riverbank*) are unrelated (homonymy), while the senses of ‘head’ share meaning similarities (polysemy). This higher semantic relatedness leads to greater contextual overlap for ‘head’, whereas ‘bank’ exhibits lower contextual overlap.

- (24) bank:
- a. I deposited money at the **bank**. SENSE: *financial institution*
  - b. I fished from the **bank**. SENSE: *riverbank*
- (25) head:
- a. I hit my **head** on the doorframe. SENSE: *body part*
  - b. I was appointed as **head** of the department. SENSE: *leadership*
  - c. The **head** of the hammer hit the floor. SENSE: *top part*

Contextual overlap alone can explain the observed performance differences between lexical ambiguity types in previous studies. Low contextual overlap suggests that context provides clearer cues for different senses, whereas high contextual overlap suggests that contextual information is less informative for ambiguity resolution. This gives polysemous senses an inherent disadvantage.

However, more importantly, sense-irrelevant features may affect embedding similarity differently across ambiguity types. Polysemous senses may be more affected because their contexts are not diverse enough to make sense-relevant features determine embedding similarity. Alternatively, sense-irrelevant features may determine embedding similarity even when contexts provide clear cues, as in the case of homonymy, causing difficulties for both ambiguity types. This would indicate that sense-irrelevant features pose a fundamental challenge for ambiguity resolution regardless of ambiguity type. Investigating this question is therefore essential for understanding the limitations of contextualized word embeddings for lexical ambiguity resolution.

**Inherent Polysemy.** Inherent polysemy, however, presents a more distinct challenge. As discussed in Section 1.2.1, inherently polysemous nouns like ‘book’ have multiple meaning facets (INFORMATION and PHYSICAL\_OBJECT) that can be simultaneously active in copredication contexts, where different predicates select different facets. For instance, in (26-a), both facets of the noun are selected by different predicates: ‘lift’ selects the PHYSICAL\_OBJECT facet, while ‘understand’ selects the INFORMATION facet. By contrast, the predicates in (26-b) and (26-c) select only one facet.

- (26) a. **Both facets (copredication):**  
I *understood* and *lifted* the **book**.
- b. **information only:**  
I *understood* the *informative* **book**.
- c. **physical\_object only:**

I *lifted* the *heavy* **book**.

It remains unclear whether contextualized word embeddings encode sufficient information for distinguishing copredication contexts, as in (26-a), from single-facet contexts, as in (26-b) and (26-c). Context can provide clear signals for detecting copredication, as copredication happens when different predicates select different facets of the same noun. However, the challenge lies in detecting the *absence* of copredication — determining when only one facet is selected. Here, the challenge is fundamentally linguistic. With inherently polysemous nouns, both facets are part of the word’s meaning regardless of context and are always potentially available even in single-facet contexts — the other facet is not absent but merely unselected. This means that the distinction between single-facet and copredication contexts is inherently subtle. This distinction may be equally challenging for humans. Therefore, understanding whether embeddings can capture these fine-grained semantic distinctions is an important question.

### 1.3.2 Creating Invariant Embeddings

In the previous section, we argued that certain contextual features can harm unsupervised lexical ambiguity resolution due to how they affect embedding similarity. Two critical questions arise from this. First, how can we create embeddings that do not encode sense-irrelevant features? Second, would removing these features improve performance? Invariant representation learning specifically addresses this type of problem. It aims to create representations that do not encode certain unwanted features of data that can harm task performance. It is widely adopted in Computer Vision (Khotanzad and Hong, 1990; Bronstein et al., 2003; Louizos et al., 2015; Jaiswal et al., 2019) and has applications in Natural Language Processing (NLP) (Louizos et al., 2017; Jaiswal et al., 2018; Peng and Zhang, 2020; Xin et al., 2022), however, mostly focusing on creating domain-invariant representations.

The unwanted features of the data can be nuisances, biases, or domain-specific features. Nuisances are features that have little or no relevance to the task but influence performance, such as facial expressions in face recognition (Bronstein et al., 2003) or orientation in image recognition (Khotanzad and Hong, 1990). Contextualized word embeddings encode various types of features as a result of their pre-training with language modeling objective. This objective optimizes for word prediction and this does not guarantee that all encoded features of the data are relevant or beneficial to downstream tasks. In this thesis, we define sense-irrelevant features as nuisances for lexical ambiguity resolution: features that affect embedding similarity and task performance despite having little or no relevance to senses. For instance, as illustrated in the previous section with (23), the syntactic role of the words can negatively affect lexical ambiguity resolution performance if embedding similarity is determined by this feature. Therefore, creating embeddings that do not encode sense-irrelevant features could improve unsupervised lexical ambiguity resolution performance.

However, it is important to note that some features may be sense-relevant for certain words but still sense-irrelevant for the majority of the words. In such cases, they can still negatively impact overall performance. Grammatical number is one such example. For instance, the singular and plural forms of ‘glass’ (‘glass’ vs. ‘glasses’) correspond to different senses. The *eyewear* sense of ‘glass’ is only expressed in the

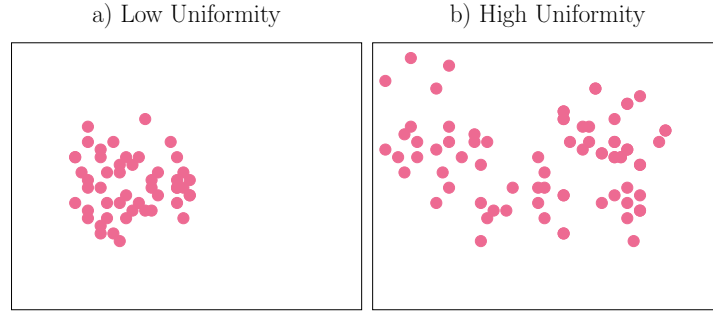


Figure 1.7: Two hypothetical embedding spaces with low (a) and high (b) uniformity.

plural form (‘glasses’), whereas the singular form (‘glass’) refers to the material or the drinking vessel. In this case, grammatical number aids ambiguity resolution; however, such cases are rare. Therefore, removing these features can be expected to improve overall task performance.

Although invariant representation learning is well-studied, its application to linguistic features in NLP remains unexplored. Moreover, it has been mainly used to pre-train encoder models (Louizos et al., 2017; Jaiswal et al., 2018, 2020). It is a promising but unexplored approach for removing sense-irrelevant features from contextualized word embeddings of PLMs while preserving their word prediction capabilities. Developing a method to remove sense-irrelevant features from the embeddings can enable us to improve contextualized word embeddings for unsupervised lexical ambiguity resolution and to investigate the relation between these features and task performance.

### 1.3.3 Creating Sense-Enhanced Embeddings

Given that contextualized word embeddings do not naturally align with senses (Yenicecik et al., 2020), an important question arises: how can we enhance these embeddings to improve their *sense-alignment*? We define sense-alignment as the degree to which instances sharing the same sense are closer together in the embedding space. One potential approach is *sense knowledge enhancement through fine-tuning*: incorporating external sense knowledge into pre-trained embeddings. Fine-tuning allows leveraging the rich linguistic knowledge already captured by PLMs without the need to train a language model for the purpose from scratch. However, fine-tuning introduces potential trade-offs in expressiveness of the embeddings.

**The Trade-off Between Sense-Alignment and Expressiveness.** Fine-tuning a PLM to improve sense-alignment may come at the cost of embedding *expressiveness*, as associated with topological properties such as *isotropy* and *uniformity* (Gao et al., 2021; Rajaei and Pilehvar, 2021; Zhang et al., 2022). Fine-tuning has been shown to reduce expressiveness by decreasing isotropy (Zhang et al., 2022).

For unsupervised lexical ambiguity resolution, isotropy and uniformity may be particularly important as unsupervised methods rely on the similarity between the embeddings, and these properties have a direct effect on embedding similarity. Both anisotropy and low uniformity can cause artificially high similarity even when there is no meaningful relation between the instances. For instance, the anisotropic na-

ture of the PLM embedding space results in high similarity between the embeddings of random words (Ethayarajh, 2019). Similarly, a less uniform distribution of embeddings may cause similar effects, because if the embeddings are concentrated in certain regions of the space, they can have artificially high similarity. An illustration of two hypothetical embedding spaces with low and high uniformity is provided in Figure 1.7.

However, this effect has not been directly measured in the context of lexical ambiguity resolution. It remains unclear which topological properties are most critical for lexical ambiguity resolution performance, and whether isotropy and uniformity play an important role in the success of sense-enhanced embeddings. Furthermore, the relation between fine-tuning for sense knowledge enhancement and its effect on topology remains unexplored. With sense knowledge enhancement, our goal is to improve the sense-alignment of the embedding space so that embedding similarity better reflects sense distinctions. However, in order to create successful sense-enhanced embeddings, improving sense-alignment alone may not be sufficient if it comes at the cost of other properties such as isotropy and uniformity.

**Fine-Tuning Methods for Sense Knowledge Enhancement.** Two fine-tuning methods are widely used with PLMs: *Contrastive Learning (CL)* and *Supervised Predictive Learning (SPL)*. Studies comparing SPL and Supervised Contrastive Learning (SCL) have primarily focused on performance differences, showing better performance with SCL in various contexts, but without investigating the quality of the resulting embeddings in terms of their topological properties (Gunel et al., 2020; Khosla et al., 2020). Moreover, studies reporting improvements with fine-tuning for sense knowledge enhancement typically employ methods beyond SPL alone (Garí Soler and Apidianaki, 2020; Mosolova et al., 2024), leaving open the questions of whether SPL can achieve comparable results and how it differs from SCL in terms of its effect on the topology of the embedding space of PLMs. While CL has been shown to improve isotropy and uniformity of the PLMs (Gao et al., 2021; Liu et al., 2021a,b; Yan et al., 2021), these studies have mostly focused on Unsupervised Contrastive Learning (UCL) and do not involve sense knowledge enhancement.

These gaps in the literature leave several critical questions underexplored. It remains unclear whether fine-tuning can successfully create sense-enhanced embeddings with SPL alone, given that previous studies have relied on methods beyond SPL. Furthermore, it remains unclear how different fine-tuning methods differ in their effect on the topology of the PLM embedding space. Overall, understanding the relation between fine-tuning methods, topological properties, and task performance is crucial for developing successful approaches to sense knowledge enhancement.

## 1.4 Research Goals, Questions, and Hypotheses

This thesis adopts both theoretical and practical perspectives, which are complementary. From a theoretical perspective, the goal is to understand how contextualized word embeddings of Pre-trained Transformer Language Models (PLMs) encode and organize sense information. From a practical perspective, based on these theoretical insights and identified limitations, the goal is to develop methods to improve their effectiveness for unsupervised lexical ambiguity resolution.

To this end, we first assess whether these embeddings encode sufficient information to detect complex lexical ambiguity phenomena in supervised settings, specifically copredication with inherently polysemous nouns. Next, due to the importance of embedding similarity in unsupervised settings, we examine which contextual features determine embedding similarity between word instances and whether they are sense-relevant. Furthermore, we investigate how sense-irrelevant features affect embedding similarity across ambiguity types with different levels of contextual overlap. As a final theoretical goal, we investigate how topological properties of the embeddings, more specifically sense-alignment, isotropy, and uniformity, relate to task performance and how different fine-tuning methods for sense knowledge enhancement affect embedding space topology differently.

These theoretical insights reveal the properties of the embeddings that limit unsupervised performance. In terms of practical goals, we address these limitations in two ways. First, we develop a method to remove sense-irrelevant features from contextualized word embeddings. Second, we propose and evaluate different fine-tuning methods for sense knowledge enhancement using Supervised Predictive Learning (SPL) and Supervised Contrastive Learning (SCL). We particularly focus on optimizing embedding space topology to improve task performance. These research questions and hypotheses guide both the theoretical and practical investigations of this thesis:

- **RQ1:** Do contextualized word embeddings encode sufficient information for detecting copredication? How difficult is copredication identification for expert human annotators?
  - **H1a:** Contextualized word embeddings encode sufficient information for copredication detection and can serve as effective input features to classifiers.
  - **H1b:** If contextualized word embeddings encode sufficient information for copredication detection (H1a), the embeddings of multilingual PLMs should enable zero-shot transfer across languages.
  - **H1c:** Because both meaning facets are always available with inherently polysemous nouns, distinguishing contexts where only one facet is active from contexts where both are simultaneously active (copredication) is challenging even for expert human annotators, as reflected in inter-annotator agreement scores.
- **RQ2:** Which contextual features determine embedding similarity between word instances, and are they sense-relevant? Do sense-irrelevant features affect embedding similarity differently across homonymous and polysemous senses?
  - **H2a:** The features determining embedding similarity between word instances are not consistently sense-relevant.
  - **H2b:** Sense-irrelevant features primarily determine embedding similarity in the case of polysemy due to high contextual overlap. In contrast, sense-relevant features determine embedding similarity in the case of homonymy due to low contextual overlap.

- **RQ3:** How can we successfully remove sense-irrelevant features from contextualized word embeddings of PLMs? What impact does this removal have on unsupervised lexical ambiguity resolution performance?
  - **H3a:** Sense-irrelevant features can be successfully removed from contextualized word embeddings, while preserving their overall embedding quality.
  - **H3b:** Although some sense-irrelevant features may correlate with senses for some words, overall they introduce noise. Therefore, removing these features will improve performance despite exceptions at the individual word level.
- **RQ4:** Can successful sense-enhanced embeddings be created through fine-tuning? How do topological properties of the embeddings (sense-alignment, isotropy, uniformity) relate to task performance? How do SCL and SPL differ in their effects on the topology of the PLM embedding space?
  - **H4a:** Successful sense-enhanced embeddings can be created through fine-tuning PLMs with both SPL and SCL.
  - **H4b:** Improving isotropy and uniformity alongside sense-alignment leads to better performance in lexical ambiguity resolution.
  - **H4c:** SCL and SPL differ in their effect on the topology of the embedding space of PLMs, with SCL expected to achieve a better balance among sense-alignment, isotropy, and uniformity for performance.

## 1.5 Summary of the Research Papers

The previous section identified the research goals and questions investigated in this thesis. To address these research questions, we present five research papers. For convenience, each paper is referred to by its short title throughout this thesis. The papers are presented in Chapters 2–6 and the following subsections summarize each paper and provide its reference and author contributions:

- **Section 1.5.1: Cross-Lingual Copredication Detection** (Yavas et al., 2023) (Chapter 2)
- **Section 1.5.2: Copredication Corpus Annotation** (Chen et al., 2025b) (Chapter 3)
- **Section 1.5.3: Feature Relevance for WSD** (Yavas, 2024) (Chapter 4)
- **Section 1.5.4: Feature Removal with Adversarial Forgetting** (Yavas et al., 2024) (Chapter 5)
- **Section 1.5.5: Sense Knowledge Enhancement through Fine-Tuning** (Yavas et al., 2025) (Chapter 6)

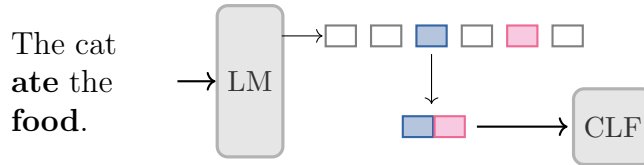


Figure 1.8: Predication classification method. The classifiers are trained to classify the relation between a predicate and its argument in a sentence using their contextualized word embeddings from a PLM as input.

### 1.5.1 Cross-lingual Copredication Detection (Chapter 2)

*Cross-lingual Copredication Detection* (Yavas et al., 2023) addresses the first part of RQ1. The paper aims to assess whether contextualized word embeddings encode sufficient information for detecting copredication with inherently polysemous nouns. For instance, (27) is a copredication example with FOOD•EVENT noun ‘dinner’, where the noun is targeted by both a predicate selecting its FOOD facet (‘vegetarian’) and a predicate selecting its EVENT facet (‘organize’).

(27) We *organized* a *vegetarian* **dinner**.

This paper develops and evaluates an automatic copredication detection method that uses contextualized word embeddings of Pre-trained Transformer Language Models (PLMs). This method utilizes classifiers that decide whether a predicate selects a specific semantic type in a given context. The paper trains individual binary classifiers for both *verbal predications* and *adjectival predications* for five semantic types: INFORMATION, EVENT, ARTIFACT, HUMAN, and LOCATION. These classifiers are implemented as support vector machines and use contextualized word embeddings of both the predicate and its argument in a sentence as input, as illustrated in Figure 1.8.

The classifiers are trained for Italian using the embeddings of Italian BERT<sup>3</sup>, due to the availability of Italian training data through Typed Predicate Argument Structures (T-PAS). For verbal predications, T-PAS is used as the resource for both training and test data. T-PAS provides corpus-derived argument structure patterns for Italian verbs with manually annotated semantic argument types, e.g., [Human] mangiare [Food] (Eng.: [Human] eat [Food]). Each verb pattern has matching corpus instances extracted from the itWac corpus (Baroni et al., 2009). For each semantic type, those instances in which the verb takes a direct object with the target semantic type or its subtype are treated as positive samples, and the remaining instances as negative samples.

Training data for adjectival predications is created through data augmentation using the verbal predication data as a foundation due to the unavailability of annotated data. For this purpose, masked word prediction with BERT is used. A mask is inserted after the target noun in verbal predication sentences, and BERT is used to predict an adjective instead of the mask as illustrated in (28). These generated sentences are automatically assigned the same semantic type label as their source verbal predication sentences, on the assumption that inserting an adjective preserves the selected semantic type in the sentence. Since this data is automatically generated, it is used for training only. The test data for adjectival predications is constructed by

<sup>3</sup>Model: dbmdz/bert-base-italian-base-cased

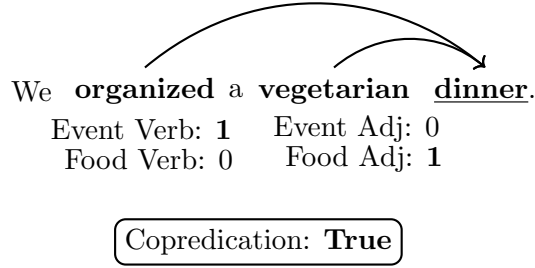


Figure 1.9: Copredication detection method. Given a sentence where both a verb and an adjective target the same noun, the method uses classifiers to detect whether these predicates select different semantic types.

extracting corpus instances from the itWac corpus, identified through a concordance search.

- (28) Bollire la **pasta** [MASK]. → Bollire la **pasta fresca**.  
*‘Boil the pasta.’* *‘Boil the **fresh** pasta.’*

The paper further investigates whether semantic type information encoded in contextualized word embeddings transfers across languages without additional training. For this purpose, multilingual classifiers are trained using the embeddings of multilingual models mBERT and XLM-RoBERTa. These classifiers are trained on Italian data, then applied in a zero-shot setting to English, German, Turkish, and Chinese. The classifiers are tested on a multilingual dataset containing five languages, where the test data for English, German, Turkish, and Chinese is derived from the Italian test data through automatic translation and subsequent manual correction.

For copredication detection, the paper proposes a method using the individual semantic type classifiers. The copredication detection method works as follows: individual classifiers are applied to the predicates that target the same noun in a sentence, and copredication is detected when predicates are classified as selecting different semantic types. The paper focuses specifically on *verb+adjective* copredication constructions with FOOD•EVENT nouns (e.g., ‘dinner’, ‘lunch’, ‘meal’), as in (27). For instance, if a verb is classified as selecting EVENT and an adjective as selecting FOOD on the same noun, copredication is detected. This process is illustrated in Figure 1.9. Since the original training data of the classifiers lacks examples with inherently polysemous nouns, it is extended with additional copredication and single-facet predication examples of FOOD•EVENT nouns, and the classifiers are re-trained on the extended data to be used for copredication detection.

For the evaluation of the method, the paper constructs a multilingual copredication test set including five languages: Italian, English, German, Turkish and Chinese. The test data for Italian is manually created and the test data for the other languages is derived from the Italian test data through automatic translation and subsequent manual correction. Copredication detection performance on all languages is tested using the multilingual classifiers, more specifically EVENT and ARTIFACT verbal and adjectival classifiers.

The results demonstrate strong performance across all semantic type classifiers for both predicate types in both monolingual and cross-lingual settings. Furthermore, copredication detection results are promising across all languages. Together,

these findings suggest that contextualized word embeddings encode sufficient information for detecting this complex phenomenon and that this information transfers successfully across languages in zero-shot settings. However, the results reveal a challenge in detecting the absence of copredication. This challenge occurs only with inherently polysemous nouns, confirming that both facets remain potentially active regardless of context and highlighting the difficulty of the task.

**Reference:** Yavas, D. E., Kallmeyer, L., Osswald, R., Ježek, E., Ricchiardi, M., & Chen, L. (2023, September). Identifying Semantic Argument Types in Predication and Copredication Contexts: A Zero-Shot Cross-Lingual Approach. In *Proceedings of the 14th International Conference on Recent Advances in Natural Language Processing* (pp. 310-320).

**Contribution:** I was responsible for implementation, conducting experiments, and writing, with contributions to methodology. Conceptualization and methodology were led by Laura Kallmeyer, who also provided supervision alongside Rainer Osswald and Elisabetta Ježek. Marta Ricchiardi contributed data annotation and preprocessing. Long Chen contributed data annotation.

### 1.5.2 Copredication Corpus Annotation (Chapter 3)

*Copredication Corpus Annotation* (Chen et al., 2025b) addresses the human performance component of RQ1. The paper details the construction and annotation of CoCoCorpus, a copredication corpus for English. The paper focuses on five nouns for each of three types of inherent polysemy: FOOD•EVENT, INFORMATION•EVENT, and PHYSICAL\_OBJECT•INFORMATION and on *verb+adjective* copredication construction.

Corpus annotation follows four steps. First, target constructions containing the selected nouns are automatically extracted from BookCorpus, specifically constructions where the target nouns are predicated by both a verb and an adjective. These candidates are then preliminarily annotated using the classifiers from the previous research paper *Cross-Lingual Copredication Detection* (Yavas et al., 2023). Then, two expert annotators manually annotate the pre-annotated examples. Finally, annotation disagreements are resolved through discussion to produce gold standard labels. Example (29) illustrates this process.

- (29) a. **Sentence Extraction:**
- Sentence:** We *organized* a *vegetarian* dinner.
  - Target noun:** ‘dinner’
  - Verb:** ‘organized’
  - Adjective:** ‘vegetarian’
- b. **Classifier Annotation:**
- ‘vegetarian’→FOOD
  - ‘organized’→EVENT
  - Copredication:** *True*
- c. **Human Annotation:**
- ‘vegetarian’→FOOD

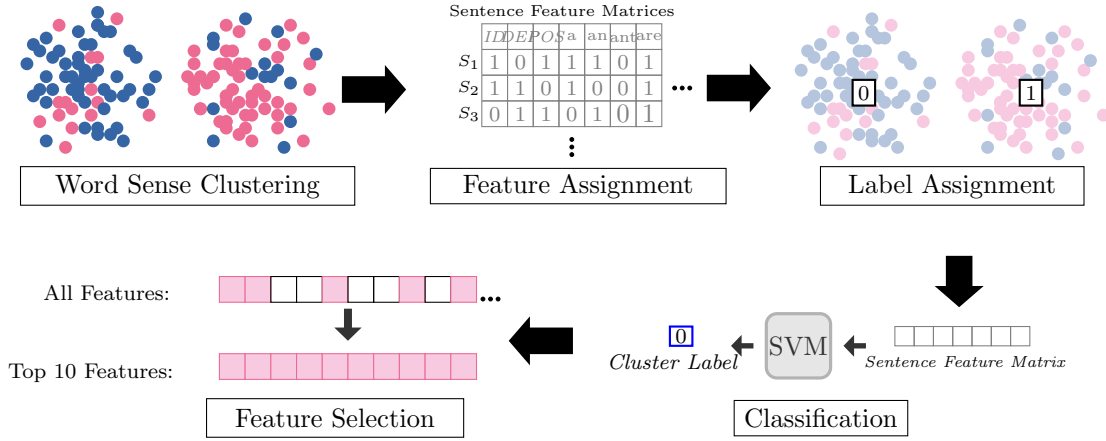


Figure 1.10: Cluster explainability method: i) Perform clustering with BERT embeddings, ii) Represent each sentence with sentence context features, iii) Assign each sentence a label based on the cluster it belongs to, iv) Train a classifier to predict the clusters that sentences belong to based on their sentence context features, v) Apply feature selection to determine the sentence context features responsible for the performance of the classifier.

- ‘organized’ → EVENT
- Copredication: *True*

d. **Gold Label:** *Copredication*

The resulting corpus contains 1,500 gold-standard annotations, including approximately 200 copredication instances. The paper further provides a detailed analysis of the annotation, including the reasons behind the annotator disagreements. The results show moderate inter-annotator agreement, with some of the disagreements stemming from unclear distinctions between the facets of inherently polysemous nouns. This demonstrates that copredication identification is challenging not only for computational models but also for human experts, contextualizing the performance of computational models on this task.

**Reference:** Chen, L., Yavas, D. E., Kallmeyer, L., & Osswald, R. (2025, September). Cococorpus: a corpus of copredication. In *Proceedings of the 21st Joint ACL-ISO Workshop on Interoperable Semantic Annotation (ISA-21)* (pp. 31-40).

**Contribution:** I was responsible for the automatic data processing, designing the annotation pipeline, manual annotation, and writing of the corresponding sections of the paper. The first author equally contributed to the design of the annotation pipeline and manual annotation, and led the remaining aspects of the paper. Conceptualization was led by Laura Kallmeyer and Rainer Osswald, who also provided supervision.

### 1.5.3 Feature Relevance for WSD (Chapter 4)

*Feature Relevance for WSD* (Yavas, 2024) addresses RQ2 by conducting a cluster explainability experiment to identify the sentence context features that drive clus-

ter formation in unsupervised Word Sense Disambiguation (WSD) and assess their relevance to distinguishing senses.

The cluster explainability method, illustrated in [Figure 1.10](#), works as follows. The target word instances are first clustered using their contextualized word embeddings from BERT. SemCor is used as the data source, with the sense labels serving as the gold standard for clustering. The method then aims to reverse the clustering process by recreating the clusters through a classification task. Each word instance is represented by its *sentence context features* and assigned a label based on its cluster. Sentence context features are the bag-of-words representation of the sentence, morphological properties, syntactic role, and position of the target word, and part-of-speech tags of the target word’s neighbors. Next, a classifier is trained to predict the cluster labels from the sentence context features. Then, *recursive feature elimination* is applied to the classifier to identify which features are most important for the classifier’s performance, revealing which sentence context features drive cluster formation. Finally, the relevance of the identified features to sense distinctions is assessed through qualitative evaluation. This process is carried out for each target word.

Additionally, the paper evaluates the usefulness of sentence context features for lexical ambiguity resolution in general. For this purpose, the paper trains another type of classifier for each target word: *classifiers for WSD*. Unlike the classifiers for cluster assignment, which aim to recreate embedding clusters, these classifiers predict sense labels directly using the same sentence context features as input.

The paper further qualitatively assesses whether sense-irrelevant features affect embedding similarity differently across homonymy and polysemy, given their different levels of contextual overlap. To distinguish homonymy from polysemy, semantic relatedness judgment scores from ([Nair et al., 2020](#)) are mapped to SemCor examples, and sense pairs are classified as homonymous or polysemous based on these scores. Additionally, the performance of the WSD classifiers is compared across homonymous and polysemous senses in order to assess the effects of contextual overlap on lexical ambiguity resolution performance.

The results show that the features driving cluster formation are mostly sense-irrelevant for both homonymy and polysemy. Unsupervised WSD performance is higher for homonymous senses than polysemous senses, and homonymous senses are more distinguishable by sentence context features. However, sense-irrelevant features determine embedding similarity for both ambiguity types, even when senses occur in highly diverse contexts with clear cues. These findings suggest that sense-irrelevant features pose a fundamental challenge for unsupervised ambiguity resolution regardless of ambiguity type.

**Reference:** Yavas, D. E. (2024). Assessing the significance of encoded information in contextualized representations to word sense disambiguation. In *Proceedings of the Third Workshop on Understanding Implicit and Underspecified Language* (pp. 42-53).

**Contribution:** Sole author

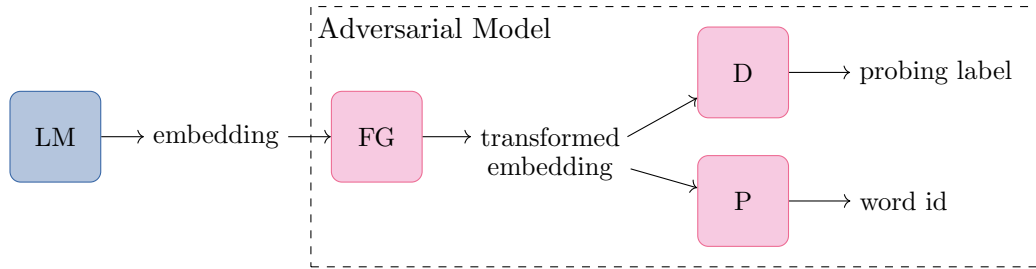


Figure 1.11: Framework for training a forget-gate ( $FG$ ) to create feature-invariant embeddings from contextualized word embeddings of a PLM. The forget-gate is trained as a part of an adversarial model with a discriminator ( $D$ ) and a predictor ( $P$ ).

### 1.5.4 Feature Removal with Adversarial Forgetting (Chapter 5)

*Feature Removal with Adversarial Forgetting* (Yavas et al., 2024) addresses RQ3 by proposing an adversarial forgetting framework to remove sense-irrelevant features from PLM embeddings. Based on the findings of *Feature Relevance for WSD* (Yavas, 2024), the paper targets two types of information: morphological and syntactic information. With the proposed framework, two types of *feature-invariant embeddings* from BERT embeddings are created: *morphology-* and *syntax-invariant embeddings*. The paper evaluates the impact of feature removal on clustering-based Word Sense Induction (WSI), comparing the performance with feature-invariant and original BERT embeddings. A word-level analysis further investigates whether feature removal is uniformly beneficial or whether the performance with certain words is harmed by the removal.

The proposed framework is an adaptation of the adversarial training framework of Jaiswal et al. (2020). It works by training a *forget-gate* that learns to transform the embeddings of a PLM to remove the targeted information, while preserving their word prediction capabilities. The forget-gate is trained as part of an adversarial model alongside a discriminator and a predictor. The training of the adversarial model follows these steps. The embeddings of the PLM are passed through the forget-gate to be transformed, and the transformed embeddings are then given as input to the discriminator and the predictor. The discriminator is trained to predict the unwanted information from these embeddings, and the predictor is trained for word prediction. The forget-gate’s weights are updated to maximize the discriminator’s loss while minimizing the predictor’s loss in order to remove the unwanted information from the embeddings while preserving their word prediction capabilities. The framework is illustrated in Figure 1.11.

The paper trains two forget-gates using the proposed framework: one targeting morphological features (for morphology-invariant embeddings) and one targeting syntactic features (for syntax-invariant embeddings). Both are trained and evaluated on the Brown corpus (Kučera et al., 1967). For each forget-gate, the discriminator is trained to probe for the specific unwanted information — predicting either morphological or syntactic features of tokens from their contextualized word embeddings. These features are annotated automatically using spaCy<sup>4</sup>. For both

<sup>4</sup><https://spacy.io/>

forget-gates, the predictor is trained to predict token identity from the embeddings.

To evaluate the forget-gates, feature-invariant embeddings are created using the trained forget-gates and two models are trained using feature-invariant embeddings as input: a word predictor and a probing model for unwanted information. The performance of these two models on the test data indicates whether the creation of feature-invariant embeddings is successful.

To evaluate the impact of feature removal on WSI performance, clustering is performed on SemCor with both types of feature-invariant embeddings and the original BERT embeddings, and their performance is compared. Furthermore, the paper analyzes the relation between the morphological and syntactic features of words and WSI performance, and how the removal of this information affects clustering performance at individual word level. For this purpose, the association between the features of word instances and their sense or cluster assignments is measured using Mutual Information. Based on Mutual Information scores, words are grouped into three cases: (1) cases where senses are associated with the targeted features, (2) cases where only BERT clusters are associated with the targeted features, and (3) cases where neither senses nor clusters are associated with the targeted features. Clustering performance is then compared between the original BERT embeddings and the feature-invariant embeddings for each case, revealing whether and for which cases feature removal improves or harms clustering performance.

The results show that the proposed framework successfully creates feature-invariant embeddings. Both types of feature-invariant embeddings yield low probing performance for the targeted unwanted information and word prediction performance close to that of the BERT embeddings. Regarding lexical ambiguity resolution performance, both morphology- and syntax-invariant embeddings improve clustering-based WSI performance over the original BERT embeddings. Furthermore, the analysis of the relation between morphological and syntactic features and WSI performance shows that morphological information mostly introduces noise, while syntactic features show stronger association with senses. However, even for words where syntactic features are associated with senses, the removal of these features causes only marginal performance drops.

**Reference:** Yavas, D. E., Bernard, T., Kallmeyer, L., & Crabbé, B. (2024). Improving word sense induction through adversarial forgetting of morphosyntactic information. In *Proceedings of the 13th Joint Conference on Lexical and Computational Semantics (\*SEM 2024)* (pp. 238-251).

**Contribution:** I was responsible for conceptualization, implementation, experiments, and writing. Methodological direction was suggested by Timothée Bernard. All co-authors provided supervision.

### 1.5.5 Sense Knowledge Enhancement through Fine-Tuning (Chapter 6)

*Sense Knowledge Enhancement through Fine-Tuning* (Yavas et al., 2025) addresses RQ4 by conducting experiments with Supervised Contrastive Learning (SCL) and Supervised Predictive Learning (SPL) fine-tuning methods for sense knowledge enhancement with both BERT and RoBERTa. It evaluates both methods in terms of

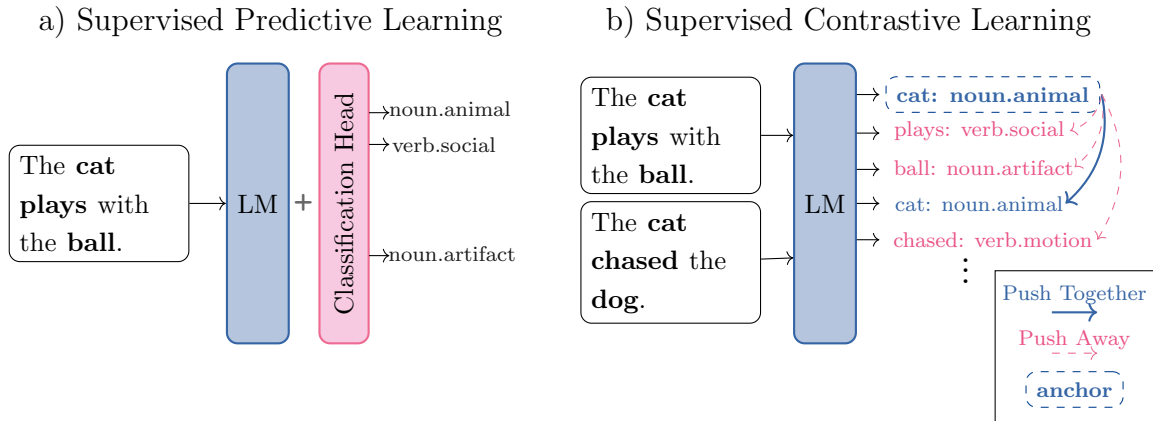


Figure 1.12: Two fine-tuning methods for sense knowledge enhancement using WordNet supersenses. For SPL (a), a supersense classification head is added to the encoder. For SCL (b), for each *anchor* (all noun and verbs instances in a batch of sentences are anchors), the encoder is fine-tuned to push closer to the anchor the word instances with the same supersense while pushing away from it those with a different supersense.

how they affect embedding space topology focusing on three topological properties: sense-alignment, uniformity, and isotropy. Furthermore, it investigates the relation between these topological properties and task performance, aiming to identify which properties are critical for creating successful sense-enhanced embeddings.

The paper enhances the embeddings of the PLMs with WordNet supersense information using different fine-tuning methods and uses the SemCor corpus as the training data. For SPL, a classification head is added to the PLM and the model is fine-tuned by minimization of the Cross-Entropy loss to predict the supersense label of each word instance (verbs and nouns). For SCL, the PLM is fine-tuned by minimization of the SCL loss (Khosla et al., 2020) to push together the embeddings of the word instances that have the same supersense label while pushing away the embeddings of the word instances with different supersense labels. See Figure 1.12 for an illustration of both methods.

The downstream task performance is evaluated on the Word-in-Context (WiC) task using both supervised and unsupervised setups. For the supervised setup, a binary classifier is trained to predict whether two instances of a word have the same meaning. The classifier takes the concatenation of their embeddings as input. For the unsupervised setup, cosine similarity between the embeddings of two word instances is calculated, and the instances are classified as having the same meaning if the cosine similarity exceeds the a certain threshold. This threshold is tuned on the development set.

To provide a more complete picture of the relation between fine-tuning methods, topology, and task performance, the paper additionally experiments with the temperature parameter of both loss functions, which influences class separation and has been shown to impact the topological properties of the embeddings with Unsupervised Contrastive Learning (UCL) (Wang and Liu, 2021). The relation between topological properties and task performance is analyzed by examining patterns across results with different embeddings. The comparison between BERT and RoBERTa further investigates whether the findings generalize across different PLMs.

In addition to these fine-tuning methods, the paper conducts further experiments to investigate the effects of two factors on task performance: *task similarity* and improving *isotropy in isolation*. For task similarity, the PLM is fine-tuned with an objective similar to the downstream task (WiC). For improving isotropy in isolation, the isotropization post-processing method proposed by [Mu and Viswanath \(2018\)](#) is applied to the original PLM embeddings, allowing the investigation of the effects of isotropy without incorporating external knowledge into the embeddings. The embeddings obtained by these methods are also evaluated on the WiC task to assess the effects of these factors on downstream task performance independently of sense knowledge enhancement.

The results show that isotropy is an important topological property for task performance. Increasing isotropy alone through post-processing improves the performance. While the embeddings with only increased sense-alignment perform worse than the original embeddings, the highest performance is achieved with embeddings that exhibit a significant increase in both isotropy and sense-alignment. In terms of the effects of uniformity, the embeddings that achieve the best performance overall balance uniformity and sense-alignment.

Furthermore, the results show that SPL and SCL produce embeddings that differ in terms of topological properties. More specifically, SCL generates embeddings that are more isotropic and more sense-aligned compared to SPL. Regarding task performance, SCL demonstrates more consistent improvements across both models, while SPL fails to create successful embeddings with BERT. Finally, the results show that fine-tuning a PLM on a task similar to the downstream task does not necessarily produce embeddings that improve downstream task performance.

**Reference:** [Yavas, D. E., Bernard, T., Crabbé, B., & Kallmeyer, L. \(2025\)](#). On the Relation Between Fine-Tuning, Topological Properties, and Task Performance in Sense-Enhanced Embeddings. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (pp. 23610-23625).

**Contribution:** I was primarily responsible for conceptualization, methodology, implementation, conducting experiments, and writing. All co-authors provided supervision.

## 1.6 Scope

This section defines the scope of this thesis, clarifying what is included and excluded from the investigation. p1-p5 refer to the five research papers introduced in the previous section:

- p1: Cross-lingual Copredication Detection ([Yavas et al., 2023](#)) ([Chapter 2](#))
- p2: Copredication Corpus Annotation ([Chen et al., 2025b](#)) ([Chapter 3](#))
- p3: Feature Relevance for WSD ([Yavas, 2024](#)) ([Chapter 4](#))

- **p4: Feature Removal with Adversarial Forgetting (Yavas et al., 2024) (Chapter 5)**
- **p5: Sense Knowledge Enhancement through Fine-Tuning (Yavas et al., 2025) (Chapter 6)**

**Pre-Trained Transformer Language Models.** We focus on BERT-based encoder models (BERT (all papers), RoBERTa (p1, p5)) and their multilingual variants (mBERT, XLM-RoBERTa) (p1). Decoder-only models such as GPT are outside our scope. We focus on encoder models because: (1) their bidirectional attention mechanism incorporates context from both directions, which is useful for lexical ambiguity resolution, and (2) they are specifically designed for language understanding rather than generation tasks.

**Pre-Trained Transformer Language Models as Feature Extractors.** We use PLMs only as feature extractors and do not fine-tune them for downstream task adaptation. Where fine-tuning is employed (p5), it serves as a tool to manipulate specific embedding properties in order to investigate their role in task performance, rather than to adapt the models to a specific task.

**Languages.** English is the primary language of investigation for most papers (p2-p5). This is due to the availability of well-established datasets and tasks for English. An exception is p1, where Italian is the primary investigation language due to the availability of resource, T-PAS. Furthermore, cross-lingual experiments in p1 investigate zero-shot transfer from Italian to English, German, Chinese, and Turkish.

**Ambiguity Types.** We examine the homonymy-polysemy distinction (p3), inherent polysemy (p1, p2), and lexical ambiguity more broadly as represented in standard datasets and tasks (p4, p5). Standard datasets do not systematically distinguish between ambiguity types. For this reason, we rely on different datasets for type-specific investigation.

**Datasets and Sense Inventories.** We use WordNet as our primary sense inventory and SemCor as our source of sense-annotated data (p3-p5), as these are the most widely used and well-established resources for English. We experiment with both fine-grained (word-specific senses) (p3, p4) and coarse-grained sense distinctions (supersenses) (p5) in WordNet. For inherent polysemy, we rely on T-PAS for semantic type annotated examples (p1), as resources for inherent polysemy and copredication are largely absent. Therefore, we construct a novel copredication detection dataset and corpus for the investigation of this phenomenon, addressing this gap (p1, p2).

**Semantic Types and Inherent Polysemy Types.** In the papers focusing on inherent polysemy and copredication (p1, p2), we focus on five semantic types: INFORMATION, EVENT, ARTIFACT, HUMAN, and LOCATION. This selection is intended to capture the diversity of the semantic type hierarchy. For inherent polysemy types, we focus on FOOD•EVENT, INFORMATION•EVENT, and PHYSICAL\_OBJECT•INFORMATION,

which are widely studied types of inherent polysemy (Pustejovsky, 1998; Ježek and Melloni, 2009; Chen et al., 2022, 2025a).

**Copredication Construction.** We focus on one copredication construction (p1, p2): *verb+adjective construction*, as in “I lifted the informative book.”.

**Improving Pre-Trained Transformer Language Models.** We focus on improving PLMs rather than pre-training new models for our purpose (p4, p5). This approach offers: (1) *computational efficiency* by eliminating costly pre-training, (2) *reduced environmental impact*, and (3) *reusability* of the resources through leveraging well-established PLMs as foundations.

**Evaluation Tasks.** We evaluate the task performance on different lexical ambiguity resolution tasks: copredication detection (p1), unsupervised WSD (p3), WSI (p4), WiC (p5). WSD, WSI, and WiC are widely used for evaluating lexical ambiguity resolution performance, and copredication detection addresses a more complex and underexplored phenomenon.

**Topological Properties.** We focus on three topological properties: sense-alignment, isotropy, uniformity (p5). Isotropy and uniformity are well-established topological properties and we complement these properties with sense-alignment to investigate how well the embedding space aligns with senses.

## 1.7 Thesis Overview

The remainder of this thesis is organized as follows. Chapters 2 through 6 present the five research papers that constitute this thesis: *Cross-Lingual Copredication Detection* (Yavas et al., 2023) (Chapter 2), *Copredication Corpus Annotation* (Chen et al., 2025b) (Chapter 3), *Feature Relevance for WSD* (Yavas, 2024) (Chapter 4), *Feature Removal with Adversarial Forgetting* (Yavas et al., 2024) (Chapter 5), and *Sense Knowledge Enhancement through Fine-Tuning* (Yavas et al., 2025) (Chapter 6). Each paper is presented in its published form, providing full methodological details, experimental results, and individual discussions. Chapter 7 synthesizes the findings, discusses the contributions and limitations, and identifies directions for future research.

## Chapter 2

# Identifying Semantic Argument Types in Predication and Copredication Contexts: A Zero-Shot Cross-Lingual Approach

# Identifying Semantic Argument Types in Predication and Copredication Contexts: A Zero-Shot Cross-Lingual Approach

Deniz Ekin Yavas<sup>1</sup>, Laura Kallmeyer<sup>1</sup>, Rainer Osswald<sup>1</sup>,  
Elisabetta Jezek<sup>2</sup>, Marta Ricchiardi<sup>3</sup>, Long Chen<sup>1</sup>

Heinrich Heine University Düsseldorf<sup>1</sup>, University of Pavia<sup>2,3</sup>

{deniz.yavas, laura.kallmeyer, rainer.osswald, chen.long}@hhu.de<sup>1</sup>,  
jezek@unipv.it<sup>2</sup>, marta.ricchiardi01@universitadipavia.it<sup>3</sup>

## Abstract

Identifying semantic argument types in predication contexts is not a straightforward task for several reasons, such as inherent polysemy, coercion, and copredication phenomena. In this paper, we train monolingual and multilingual classifiers with a zero-shot cross-lingual approach to identify semantic argument types in predications using pre-trained language models as feature extractors. We train classifiers for different semantic argument types and for both verbal and adjectival predications. Furthermore, we propose a method to detect copredication using these classifiers through identifying the argument semantic type targeted in different predications over the same noun in a sentence. We evaluate the performance of the method on copredication test data with Food•Event nouns for 5 languages.

## 1 Introduction

This paper is concerned with the question of how to automatically decide which semantic type is targeted in predications over nouns. In our case, the predicate can be a verb or an adjective. This question is particularly interesting in cases where complex type nouns<sup>1</sup> are arguments of predications. But even with nouns that, lexically, have only a single type, the predication can target a different type and thereby trigger a coercion in the noun (Pustejovsky, 1991). Examples are given in (1). In both (1-a) as well as (1-b), the respective predicates target one of the two types of a complex type noun (a *dinner* is inherently both an Event and a Food item). In (1-c), the noun is a simple type noun (*soup* is only of type Food), and its type is targeted in the predication. The predication in (1-d) involves a coercion since it targets a type that is

different from the lexical type of the noun. Finally, for complex type nouns, we can have cases where different component types of the same noun are targeted, either by different predicates as in (1-e) or by a single predicate as in (1-f) where *book* is a physical object and an informational content at the same time and the predicate targets both. The first case is an instance of copredication (see below).

- (1) a. They chose the *vegetarian dinner*.  
→ (target: *Food*)
- b. I *organized* a *dinner* for them.  
→ (target: *Event*)
- c. I *ate* my *soup*.  
→ (target: *Food*)
- d. I *finished* my *soup*.  
→ (target: *Event*, coercion)
- e. They *organized* a *vegetarian dinner*.  
→ (target: *Event* and *Food*)
- f. He *wrote* a lot of *books*.  
→ (target: *Phys\_Obj*•*Information*)

Our main goal is to develop classifiers that, given a predicate and an argument noun in their sentential context, decide whether a specific type has been targeted. Furthermore, we exploit the cross-lingual transfer potential of multilingual *pre-trained language models* (LMs) in order to apply this task to different languages without the need of labelled data for all of them.

One interesting application of such classifiers is the detection of instances of copredication with complex type nouns. Copredication is a general term defining a “grammatical construction in which two predicates jointly apply to the same argument” (Asher, 2011, p. 11). We are interested in a specific type of copredication where two predicates that require different semantic types apply to the same noun (Pustejovsky, 1995; Pustejovsky and Jezek, 2008; Asher, 2011). For example, given the occur-

<sup>1</sup>Also “dot object” nouns (Pustejovsky, 1995), “nouns with facets” (Cruse, 1995), “dual aspect nouns” (Asher, 2011) in the literature.

rence of a complex type noun such as *dinner* in a sentence where we have two (or more) predication over that noun, we want to decide whether these predication target different types, as for instance in (1-e). We will apply the classifiers developed in this paper to this task, using the complex type *Food•Event* as a test case.

We start by investigating whether it is possible to train classifiers for both verbal and adjectival predication for this purpose using LMs (BERT (Devlin et al., 2019), mBERT (Devlin et al., 2019), XLM-RoBERTa (Conneau et al., 2020a)) as feature extractors. In addition, we investigate whether it is possible to train multilingual classifiers with a zero-shot cross-lingual approach by training the classifiers on one language with the extracted embeddings of multilingual language models and applying them to other languages.

We train monolingual classifiers for Italian using the extracted embeddings of a monolingual BERT model<sup>2</sup> and the multilingual ones using the embeddings of the multilingual models mBERT and XLM-RoBERTa. We start with Italian as our source language due to the availability of annotated data in the T-PAS (Typed Predicate Argument Structures) resource (Jezek et al., 2014). We train classifiers for verbal predication for the semantic types *Human*, *Information*, *Event*, *Artifact*, and *Location* and adjectival predication for the semantic types *Event*, *Artifact*, and *Information*. The selection of these types is intended to capture the diversity of the semantic type hierarchy.

Finally, we apply the verbal and adjectival classifiers for *Artifact*<sup>3</sup> and *Event* semantic types to the sentences containing *Food•Event* nouns in order to detect certain copredication patterns, as in (1-e), in which a verb and an adjective predicate over the same noun. We evaluate the proposed model on test data for a set of typologically diverse languages; Chinese, English, German, Italian, and Turkish.<sup>4</sup>

## 2 Related Work

### 2.1 Selectional Preference and Semantic Type Knowledge of LMs

There is no study to our knowledge that aims at exploiting LMs for a selectional preference task, nor

that investigates the transferability of selectional preference knowledge to other languages using multilingual LMs. However, there are studies that investigate the LMs’ knowledge about selectional preferences of verbs and semantic types. Their findings suggest that contextual language models encode information about the selectional preferences of verbs (Metheniti et al., 2020; Chersoni et al., 2021; Li et al., 2021; Pedinotti et al., 2021) and the semantic type of the nouns in general (Zhao et al., 2020). Similar to our study, Zhao et al. (2020) and Chersoni et al. (2021) trained classifiers using the extracted representations of the BERT model for their tasks and these classifiers achieved high accuracy scores.

### 2.2 Zero-Shot Cross-Lingual Transfer using multilingual LMs

Several studies have investigated the performance of multilingual LMs for zero-shot cross-lingual transfer on a variety of tasks, e.g. NER, POS, NLI, QA. They show that these models are effective for this purpose (Pires et al., 2019; Conneau et al., 2020a; Wu and Dredze, 2020; Aghazadeh et al., 2022). It is also shown that these models perform well on multilingual benchmarks such as XTREME (Hu et al., 2020) and XGLUE (Liang et al., 2020). Additionally, these papers show that XLM-RoBERTa performs better than mBERT (Conneau et al., 2020a; Hu et al., 2020; Liang et al., 2020; Lauscher et al., 2020).

Recent research has also investigated the effects of language differences in cross-lingual transfer. It has been shown that structural similarity, such as word order or typological similarity affects the transfer (Pires et al., 2019; Conneau et al., 2020b; K et al., 2020; Lauscher et al., 2020; Deshpande et al., 2022). The difference in the language script is also shown to be important, but only when the word order differs as well (Pires et al., 2019; Deshpande et al., 2022).

### 2.3 Copredication Detection

Jezek and Vieu (2014) adopt a semi-automatic approach for extracting copredications of *Physical\_Object•Information* nouns with a verb and an adjective in Italian. First, they manually select a list of predicates for both semantic types: Physical Object and Information. Then, they construct copredication contexts with different predicate combinations and extract examples by searching the corpus for these contexts. As an extension to the

<sup>2</sup>See Appendix A for information about the models, parameters and libraries used for the experiments.

<sup>3</sup>Artifact is the supertype of Food in the T-PAS semantic type hierarchy.

<sup>4</sup>Datasets and code are available at: <https://github.com/yavasde/predication-classification>

previous study, [Vieu et al. \(2015\)](#) use a latent semantic distributional model in order to select the predicates to avoid the manual process of predicate selection. Compared to these studies, our method is automatic and does not rely on the classification of each predicate, which can be problematic due to their polysemous nature, but relies on the classification of each predication instance. This also allows using this method cross-linguistically without knowledge specific to each language.

## 3 Method

### 3.1 Predication Classifiers

We first show that it is possible to train a classifier for the identification of the semantic argument type targeted by a predicate in a specific predication context using the extracted representations of LMs. Furthermore, we aim to investigate whether this knowledge is transferable from one language to another with a zero-shot cross-lingual approach by training classifiers on the source language using the multilingual representations of the multilingual LMs and applying the trained models to the target languages.

For all semantic types and predication types, we use LMs as feature extractors and train monolingual and multilingual classifiers with the SVM algorithm using the extracted embeddings of the models.<sup>5</sup> For monolingual Italian classifiers, we use a BERT model for Italian and for multilingual classifiers, we use the multilingual LMs mBERT and XLM-RoBERTa. We train binary classifiers for each of the semantic types *Artifact*, *Event*, *Human*, *Information* and *Location* for verbal predications and *Artifact*, *Event*, and *Information* for adjectival predications.<sup>6</sup>

We use the contextualized embeddings of the predicate and the argument in a specific sentence as input for the classifiers. First, we tokenize each sentence with the model tokenizer and give the tokenized sentence as input to the model. Then, we extract the embeddings of the predicate (verb/adj) and the argument (direct object/noun) from the last 4 layers of the model output and average them to create one representation for each item.<sup>7</sup> We use

<sup>5</sup>Even though we tested several classification algorithms, we used SVM for the final experiments because it performed best. See Appendix B for the detailed comparison.

<sup>6</sup>We use binary classifiers instead of a multiclass one because there are predicates that can target both semantic types as in example (1-f).

<sup>7</sup>In the cases in which the target words are tokenized into

only the last 4 layers because higher layers are more specialized in semantics-related tasks ([Liu et al., 2019](#); [Tenney et al., 2019](#); [Zhao et al., 2020](#)). We formalize the task as a relation classification problem where we classify the relation between the predicate and its argument. For this purpose, we concatenate the embeddings of the predicate and the argument and use the final embedding as the input for the classifiers.<sup>8</sup>

### 3.2 Copredication Detection

In order to detect copredication, the classifiers are applied to the sentences with complex type nouns, where both syntactic types of predications are available for the same noun. First, sentences are parsed using the Stanza library ([Qi et al., 2020](#)) in order to identify the predications in sentences. Then, the embeddings of the predicate-argument pairs are extracted from the LM, concatenated and given to the relevant classifiers (verb/adj). Copredication is considered detected if both verbal and adjectival predication classifiers of different semantic types classify the predications as positive.

## 4 Training Classifiers

### 4.1 Data

#### 4.1.1 Verbal Predication Classifiers

**Training data.** We use T-PAS (Typed Predicate Argument Structures; [Jezek et al., 2014](#)) as our primary resource. T-PAS provides corpus-derived argument structure patterns for Italian verbs with manually annotated semantic argument types; e.g. [Human] mangiare [Food] (*Eng.:* [Human] eat [Food]). Each verb pattern has matching corpus instances extracted from the itWac corpus ([Baroni et al., 2009](#)).

In T-PAS, semantic types are organized in a hierarchy. For each semantic type (*Human*, *Event*, *Information*, *Artifact*, *Location*), we extract sentences whose verbs take direct objects with the target semantic type or a subtype of it.

The training negatives are also selected from T-PAS from the semantic types other than the target semantic type’s supertypes, subtypes, or the semantic type itself. The negatives are downsized to make their size equal to the positive samples.

subwords by the model tokenizer, only the first subword is taken into account.

<sup>8</sup>Fine-tuning is the most standard way to use LMs for token or sentence classification but it is not that straightforward to fine-tune the models for relation classification.

| Types                  | Data Size |      |     |      |     |     | Model | Languages   |             |             |             |             | Avg.               |
|------------------------|-----------|------|-----|------|-----|-----|-------|-------------|-------------|-------------|-------------|-------------|--------------------|
|                        | Training  |      |     | Test |     |     |       | it          | de          | en          | tr          | zh          |                    |
| Verbal Predication     |           |      |     |      |     |     |       |             |             |             |             |             |                    |
| Arti.                  | 522       | 258  | 248 | 236  | 220 | 182 | B     | 0.95 (0.94) | -           | -           | -           | -           | -                  |
|                        |           |      |     |      |     |     | mB    | 0.92 (0.90) | 0.84 (0.75) | 0.92 (0.91) | 0.75 (0.74) | 0.83 (0.87) | 0.85 (0.83)        |
|                        |           |      |     |      |     |     | XR    | 0.90 (0.92) | 0.86 (0.83) | 0.93 (0.92) | 0.88 (0.84) | 0.92 (0.91) | <b>0.89</b> (0.88) |
| Event                  | 643       | 317  | 258 | 268  | 276 | 256 | B     | 0.95 (0.95) | -           | -           | -           | -           | -                  |
|                        |           |      |     |      |     |     | mB    | 0.94 (0.93) | 0.88 (0.90) | 0.94 (0.93) | 0.86 (0.88) | 0.90 (0.89) | 0.90 (0.90)        |
|                        |           |      |     |      |     |     | XR    | 0.94 (0.95) | 0.89 (0.88) | 0.94 (0.93) | 0.91 (0.88) | 0.91 (0.92) | <b>0.91</b> (0.91) |
| Hum.                   | 292       | 144  | 130 | 128  | 126 | 74  | B     | 0.92        | -           | -           | -           | -           | -                  |
|                        |           |      |     |      |     |     | mB    | 0.92        | 0.93        | 0.98        | 0.84        | 0.89        | 0.91               |
|                        |           |      |     |      |     |     | XR    | 0.94        | 0.98        | 0.98        | 0.96        | 0.94        | <b>0.96</b>        |
| Info.                  | 176       | 88   | 82  | 86   | 86  | 70  | B     | 0.98        | -           | -           | -           | -           | -                  |
|                        |           |      |     |      |     |     | mB    | 0.98        | 0.80        | 0.98        | 0.84        | 0.90        | 0.90               |
|                        |           |      |     |      |     |     | XR    | 0.97        | 0.98        | 0.95        | 0.95        | 0.97        | <b>0.96</b>        |
| Loc.                   | 321       | 159  | 148 | 148  | 142 | 132 | B     | 0.95        | -           | -           | -           | -           | -                  |
|                        |           |      |     |      |     |     | mB    | 0.92        | 0.91        | 0.94        | 0.71        | 0.93        | 0.88               |
|                        |           |      |     |      |     |     | XR    | 0.95        | 0.97        | 0.97        | 0.89        | 0.93        | <b>0.94</b>        |
| Adjectival Predication |           |      |     |      |     |     |       |             |             |             |             |             |                    |
| Arti.                  | 252       | 3680 | -   | 148  | -   | -   | B     | 0.84 (0.84) | -           | -           | -           | -           | -                  |
|                        |           |      |     |      |     |     | mB    | 0.90 (0.91) | -           | 0.93 (0.93) | -           | -           | <b>0.91</b> (0.92) |
|                        |           |      |     |      |     |     | XR    | 0.87 (0.85) | -           | 0.93 (0.92) | -           | -           | 0.90 (0.88)        |
| Event                  | 564       | 1676 | -   | 148  | -   | -   | B     | 0.88 (0.89) | -           | -           | -           | -           | -                  |
|                        |           |      |     |      |     |     | mB    | 0.86 (0.88) | -           | 0.76 (0.77) | -           | -           | 0.81 (0.82)        |
|                        |           |      |     |      |     |     | XR    | 0.81 (0.82) | -           | 0.84 (0.83) | -           | -           | <b>0.82</b> (0.82) |
| Info.                  | 132       | 2536 | -   | 78   | -   | -   | B     | 0.91        | -           | -           | -           | -           | -                  |
|                        |           |      |     |      |     |     | mB    | 0.90        | -           | 0.94        | -           | -           | <b>0.92</b>        |
|                        |           |      |     |      |     |     | XR    | 0.91        | -           | 0.89        | -           | -           | 0.90               |

Table 1: The data size and the test results of each classifier. F1 scores are given. The results of the cross-linguistically best-performing classifiers are given in bold. *T-PAS+CT* results are given in parentheses.

To this end, the sentences are clustered with K-Means algorithm using the Scikit-learn library and an equal number of sentences are selected from each cluster. This undersampling method is chosen to have a balanced representation of the negatives.

The selected sentences for both positives and negatives are parsed with the spacy-udpipe Python library<sup>9</sup> in order to identify and annotate the verb and the direct object in each sentence.<sup>10</sup>

**Cross-lingual test data.** The test data is selected by splitting the data (test size %33) extracted from T-PAS. The data are then machine translated using DeepL API<sup>11</sup> to the other languages.

It is required that the verbs and objects are correctly identified in the translations. For this purpose, they are translated out-of-context and searched for in the sentences. Additionally, the translations are parsed using the Stanza library and all the verb-object pairs in the sentences are extracted through their dependency labels in order to find the correct pairs. However, sometimes, the pairs are not found automatically, in which case they are manually annotated.

In a final step, all translated sentences are manually checked and corrected by (near-)native speak-

ers of the respective languages following the guideline presented in Appendix C. Sentences that can not be corrected are eliminated. Equal numbers of negatives and positives are selected for each dataset. The resulting data numbers for each language are given in Table 1.

#### 4.1.2 Adjectival Predication Classifiers

**Training data.** The training data for the adjectival predication classifiers are generated using *Masked Language Modeling* (MLM) with BERT due to the unavailability of annotated data. We generate data for 3 semantic types *Artifact*, *Event* and *Information* using the verbal predication datasets for these types as the basis. We insert an adjective that is predicted by the model into the sentences in order to modify the direct object. The assumption is that in sentences where the verbal predication over the objects targets a certain type, the adjectives predicted by the model with a high probability score will do so as well.

First, a mask is inserted after the noun, and then the Italian BERT is made to predict a word instead of the mask. Only word predictions over a certain confidence score (0.15) are selected from the model predictions. For the final step, the predicted word is inserted in place of the mask and the resulting sentence is parsed with the spaCy library<sup>12</sup> to check

<sup>9</sup>Available at: <https://spacy.io/universe/project/spacy-udpipe>

<sup>10</sup>Sentence-level annotations are not provided in T-PAS.

<sup>11</sup>Available at: <https://www.deepl.com/>

<sup>12</sup>Available at: <https://spacy.io/>

if the relation between the noun and the predicted word is the desired one (*adjectival modification*). The sentences that meet these conditions are used for the training of the classifiers.<sup>13</sup>

**Cross-lingual test data.** Since the adjective data is generated, we do not test the performance of the classifiers on this data but on manually constructed data for Italian and English.

The test data for Italian are created by extracting corpus instances from the itWac corpus, identified through a concordance search for the most typical 5-10 lexical items that express each type in corpus instances and their respective most frequent adjectival modifiers. The sentences are extracted for 3 semantic types (*Artifact*, *Event* and *Information*) and the negatives of the test data are selected from the sentences of the other 2 semantic types.

The test data for English are also constructed by extracting corpus examples. First, good representatives of each semantic type noun are selected based on their occurrence in the T-PAS data; these are the nouns that only occur in the target semantic type data and occur more than once. As the next step, we translate the selected nouns to English and extract sentences with these nouns from the ukWac corpus (Baroni et al., 2009) but only consider the ones where the noun is the direct object of a verb and also have a token size between 3 and 20. We parse the sentences using the Stanza library and select the ones where there is an adjective that modifies the noun. Finally, we manually select the sentences with good examples of adjectives. The semantic types are the same as for Italian and the negatives are constructed similarly.

Both the test and training data are balanced in terms of the number of positives and negatives. The data size for adjectival predication classifiers can be seen in Table 1.

## 4.2 Experiments and Results

We test the monolingual classifiers on Italian test data ('B' for monolingual Italian BERT based classifiers) and the multilingual classifiers on the cross-lingual test data ('XR' for XLM-RoBERTa and 'mB' for mBERT based classifiers). F1 score is used as the metric and cross-lingual performance is evaluated by comparing the average f1 score on cross-lingual test data, see Table 1 for the re-

<sup>13</sup>The original adjectives in the sentences are replaced by model-predicted ones, in order to avoid copredication instances in the training data.

sults. The detailed results with precision and recall scores can be found in the Appendix D. A language-specific evaluation is given in Appendix E.

**Overall results.** The monolingual classifiers perform very well on the task. Each monolingual verbal predication classifier achieves over 0.92 f1 score and each monolingual adjectival predication classifier achieves over 0.84 f1 score. Similarly, all multilingual verbal predication classifiers achieve over 0.85 average f1 scores for all languages and all multilingual adjectival predication classifiers achieve over 0.81 average f1 scores for English and Italian. Overall, XR-classifiers perform better than mB-classifiers (See Table 1).

**Monolingual vs. multilingual.** The comparison of the monolingual and multilingual classifiers' performances on Italian test data shows that on average, the monolingual classifiers perform better than the multilingual ones on the source language test data. However, for some semantic types, such as *Human* (verb) and *Artifact* (adj), XR-classifiers perform better than the monolingual classifiers. (See Table 1 for the individual results and Figure 1 for the average for verbs.)

**Verbal vs. adjectival predications.** Overall, the performance of the verbal predication classifiers is better than the adjectival predication classifiers. Contrary to verbal predication classifiers, mB-classifiers perform better than XR-classifiers for adjectives overall. However, the performance difference is smaller.

## 5 Copredication Detection

### 5.1 Classifiers for Complex Type Nouns

Even though T-PAS is not necessarily a resource with simple type nouns, the number of sentences with Food•Event nouns is low in our datasets.<sup>14</sup> Since our task is to detect copredication with complex type nouns, we require classifiers that can disambiguate the meanings of these nouns.

In order to address this, we add, to each classifier's training data, additional data with complex type nouns, in which only one type of the noun is targeted; Food or Event as in (1-a) and (1-b). We add the additional data to the training data of Artifact and Event classifiers for both verbs and adjectives. The original classifiers will be referred to as 'T-PAS' and the latter as 'T-PAS+CT'.

<sup>14</sup>There are 2 sentences in the Artifact and 3 in the Event dataset.

**Training data with complex type nouns.** Additional training is obtained by extracting the sentences of Food•Event nouns with Food or Event predications from corpus. First, we determine the best predicates for each type of predication; best food verbs, event adjectives, etc. For this, we use our datasets. We extract the predicates from each semantic type dataset (Artifact and Event) and select the predicates that occur more than once and that only occur in the target semantic type dataset. In the second step, we select 9 Food•Event nouns (see Appendix F for the selected nouns) and we extract the sentences of these nouns with the selected predicates from the itWac corpus. Finally, we add the complex type sentences both to the positives and negatives of Artifact and Event training data for verbs and adjectives with the amount of 20%.

**Training results.** The performance of the T-PAS+CT classifiers on the test data can be seen in Table 1. Their performance is close to the T-PAS classifiers overall, with some slight differences for some semantic types and languages.

## 5.2 Evaluation

We apply both semantic type classifiers (Artifact and Event) to classify the verbal and adjectival predications in the sentences of the test data. We investigate how often copredication is detected both in the positives and negatives of the test data. However, we do not consider the correct classification of individual predications in this evaluation method. We use an additional evaluation method to investigate how often the predications are identified correctly.

We test both T-PAS and T-PAS+CT classifiers on the cross-lingual copredication test data comprising 5 languages; Chinese, English, German, Italian, and Turkish. We use monolingual classifiers for Italian and XR-classifiers for other languages since they performed better on single predication classification overall.

Additionally, we investigate the effects of the complex type nouns on copredication detection. We do that by comparing the performance of the method on two types of negatives: negatives with simple type nouns and complex type nouns.

### 5.2.1 Evaluation Data

The test data is manually created for Italian and machine translated into Chinese, English, German, and Turkish. The translations are manually corrected by (near-)native speakers of the respective

| Lang.           | Classifier | Scores |             |             |
|-----------------|------------|--------|-------------|-------------|
|                 |            | Sens.  | Spec.       | <i>g</i>    |
| it              | T-PAS      | 0.66   | 0.35 (0.79) | 0.48        |
|                 | T-PAS+CT   | 0.46   | 0.62 (0.87) | <b>0.53</b> |
| de              | T-PAS      | 0.66   | 0.25 (0.83) | 0.40        |
|                 | T-PAS+CT   | 0.53   | 0.58 (0.91) | <b>0.55</b> |
| en              | T-PAS      | 0.83   | 0.29 (0.79) | 0.49        |
|                 | T-PAS+CT   | 0.70   | 0.66 (0.83) | <b>0.67</b> |
| tr              | T-PAS      | 0.76   | 0.25 (0.75) | 0.43        |
|                 | T-PAS+CT   | 0.53   | 0.45 (0.87) | <b>0.48</b> |
| zh              | T-PAS      | 0.82   | 0.59 (0.83) | <b>0.69</b> |
|                 | T-PAS+CT   | 0.68   | 0.65 (0.83) | 0.66        |
| Random Baseline |            | 0.25   | 0.25        | 0.25        |

Table 2: Performance on the cross-lingual copredication test data. *g* stands for the geometric mean of specificity and sensitivity. The results of the classifiers with the best overall performance are given in bold. The specificity scores in the parenthesis refer to the specificity over simple type nouns.

languages. A similar correction procedure is applied to the test data, following the data correction guidelines in Appendix C.

The test data contains 30 positive and 24 negative examples of copredication with different semantic types (for more details, see Appendix G). In the positives, verbs and adjectives target different types of Food•Event nouns, whereas in the negatives, both predicates target the same type of Food•Event nouns (either Event or Food). An example of positives is given in (1-e), where the verb ‘organize’ targets the Event type and the adjective ‘vegetarian’ targets the Food type. As an example of the negatives, in (2-a), both the verb ‘eat’ and the adjective ‘cold’ target the Food type.

We prepare additional data for negatives with simple type nouns. We do this by substituting the Food•Event nouns in the negatives with a Food or Event simple type noun as in (2-b) (see Appendix G for more details).

- (2) a. It’s depressing to *eat* a *cold* lunch.
- b. It’s depressing to *eat* a *cold* soup.

### 5.2.2 Results

We evaluate the results using three metrics; *sensitivity* (recall), to measure the ability to detect the positives and *specificity*, to measure the ability to detect the negatives, and finally, the geometric mean of sensitivity and specificity, for the overall performance. The results can be seen in Table 2.

**Overall.** T-PAS classifiers achieve higher sensitivity scores compared to specificity scores for all languages. Even though the sensitivity scores achieve

0.80, specificity scores are around the random baseline for most of the languages. The difference between both scores is lower for Chinese and the specificity score is also good. With T-PAS+CT classifiers, there is an increase in specificity scores but also a drop in sensitivity scores for all languages. The scores for sensitivity and specificity are closer to each other. Overall, TPAS+CT classifiers perform better in terms of their overall performance for all languages except for Chinese.

**Simple type nouns.** The results of specificity scores on different types of negatives show that the low specificity score is much higher in the negatives with complex type nouns compared to the negatives with simple type nouns. The specificity scores increase with T-PAS+CT classifiers also for the second type of negatives however the difference between the two types of classifiers is much lower. For example, the increase for Italian is from 0.79 to 0.87 compared to 0.35 to 0.62.

## 6 Discussion

The findings of the previous studies suggest that LMs encode information about the selectional preferences of verbs (Metheniti et al., 2020; Chersoni et al., 2021; Li et al., 2021; Pedinotti et al., 2021) and semantic types of nouns (Zhao et al., 2020). Our study shows that it is possible to exploit this knowledge of LMs to train classifiers for the identification of the semantic types targeted by both verbs and adjectives.

From a cross-lingual point of view, our results show that it is possible to use the embeddings of the multilingual LMs to train classifiers in order to transfer knowledge from one language to another. Our results are in line with the previous studies in terms of the performance of individual models. XLM-RoBERTa yields better performance compared to other multilingual LMs (Conneau et al., 2020a; Hu et al., 2020; Liang et al., 2020; Lauscher et al., 2020) and its performance is comparable to monolingual models (Conneau et al., 2020a). Even though we have limited test data for adjectival predication classifiers, we expect the transfer to work similarly for both types of predications (verbal and adjectival) and the results for English show this is the case.

In the copredication detection task, our results show that classifiers that are trained only with data with simple type nouns are not able to disambiguate the meanings of complex type nouns. This is evi-

dent in the tendency of false positives (low specificity) with T-PAS classifiers. Even when both predications target the same semantic type in a sentence, i.e. in negatives, copredication is detected. This is because both semantic type classifiers tend to classify the predications as positive when a complex type noun is involved. However, this tendency is absent with simple type nouns, which is also evident in the specificity scores. We think that this tendency is due to the nature of complex type nouns and how they are represented by LMs, which is a topic we intend to investigate in the future. The false positive tendency is overcome by adding more data with complex type nouns and this improves the overall performance which shows that copredication detection is possible with the proposed model.

Cross-linguistically, the performance on copredication detection shows a similar pattern for all languages and for both monolingual and multilingual classifiers. In the future, we plan to use this method for building a cross-lingual collection of corpus-based copredication instances that includes also other complex types and copredication constructions.

## 7 Conclusion

In this study, we focused on training classifiers for the identification of the semantic argument types targeted by the predicates in a specific predication context using the extracted embeddings of LMs. We trained both monolingual and multilingual classifiers for different semantic types and for both verbal and adjectival predications. The training results for individual classifiers show that it is possible to train classifiers for this purpose using LMs and to train multilingual classifiers with zero-shot cross-lingual transfer using multilingual LMs. Furthermore, we proposed a method to detect copredications using these classifiers and evaluated the method’s performance on cross-lingual copredication test data. Our results show that copredication detection is a more complicated task. However, the method achieves reasonable scores for all languages and good scores for English.

## Acknowledgments

This study is a part of the project “Coercion and Copredication as Flexible Frame Composition” funded by DFG (Deutsche Forschungsgemeinschaft). We would like to thank Younes Samih for his invaluable insights and feedback.

## References

- Ehsan Aghazadeh, Mohsen Fayyaz, and Yadollah Yaghoobzadeh. 2022. [Metaphors in pre-trained language models: Probing and generalization across datasets and languages](#). In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2037–2050, Dublin, Ireland. Association for Computational Linguistics.
- Nicholas Asher. 2011. *Lexical Meaning in Context. A Web of Words*. Cambridge University Press, Cambridge.
- Marco Baroni, Silvia Bernardini, Adriano Ferraresi, and Eros Zanchetta. 2009. [The wacky wide web: a collection of very large linguistically processed web-crawled corpora](#). *Language Resources and Evaluation*, 43(3):209–226.
- Emmanuele Chersoni, Enrico Santus, Chu-Ren Huang, and Alessandro Lenci. 2021. [Decoding word embeddings with brain-based semantic features](#). *Computational Linguistics*, 47(3):663–698.
- Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020a. [Unsupervised cross-lingual representation learning at scale](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.
- Alexis Conneau, Shijie Wu, Haoran Li, Luke Zettlemoyer, and Veselin Stoyanov. 2020b. [Emerging cross-lingual structure in pretrained language models](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6022–6034, Online. Association for Computational Linguistics.
- D. Alan Cruse. 1995. [Polysemy and related phenomena from a cognitive linguistic viewpoint](#). *Computational lexical semantics*, pages 33–49.
- Ameet Deshpande, Partha Talukdar, and Karthik Narasimhan. 2022. [When is BERT multilingual? isolating crucial ingredients for cross-lingual transfer](#). In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3610–3623, Seattle, United States. Association for Computational Linguistics.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [BERT: Pre-training of deep bidirectional transformers for language understanding](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
- Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. 2020. [XTREME: A massively multilingual multi-task benchmark for evaluating cross-lingual generalisation](#). In *Proceedings of the 37th International Conference on Machine Learning*, pages 4411–4421. Proceedings of Machine Learning Research.
- Elisabetta Jezek, Bernardo Magnini, Anna Feltracco, Alessia Bianchini, and Octavian Popescu. 2014. [T-PAS; a resource of typed predicate argument structures for linguistic analysis and semantic processing](#). In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC’14)*, pages 890–895, Reykjavik, Iceland. European Language Resources Association (ELRA).
- Elisabetta Jezek and Laure Vieu. 2014. [Distributional analysis of copredication: towards distinguishing systematic polysemy from coercion](#). In *1st Italian Conference on Computational Linguistics (CLiC-it 2014)*, volume 1, pages 219–223, Pisa, Italy.
- Karthikeyan K, Zihan Wang, Stephen Mayhew, and Dan Roth. 2020. [Cross-lingual ability of multilingual BERT: an empirical study](#). In *8th International Conference on Learning Representations (ICLR’20)*, Addis Ababa, Ethiopia.
- Anne Lauscher, Vinit Ravishankar, Ivan Vulić, and Goran Glavaš. 2020. [From zero to hero: On the limitations of zero-shot language transfer with multilingual Transformers](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4483–4499, Online. Association for Computational Linguistics.
- Bai Li, Zining Zhu, Guillaume Thomas, Yang Xu, and Frank Rudzicz. 2021. [How is BERT surprised? layerwise detection of linguistic anomalies](#). In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4215–4228, Online. Association for Computational Linguistics.
- Yaobo Liang, Nan Duan, Yeyun Gong, Ning Wu, Fenfei Guo, Weizhen Qi, Ming Gong, Linjun Shou, Daxin Jiang, Guihong Cao, Xiaodong Fan, Ruofei Zhang, Rahul Agrawal, Edward Cui, Sining Wei, Taroon Bharti, Ying Qiao, Jiun-Hung Chen, Winnie Wu, Shuguang Liu, Fan Yang, Daniel Campos, Rangan Majumder, and Ming Zhou. 2020. [XGLUE: A new benchmark dataset for cross-lingual pre-training, understanding and generation](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6008–6018, Online. Association for Computational Linguistics.
- Nelson F. Liu, Matt Gardner, Yonatan Belinkov, Matthew E. Peters, and Noah A. Smith. 2019. [Linguistic knowledge and transferability of contextual representations](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Lan-*

- guage Technologies, Volume 1 (Long and Short Papers), pages 1073–1094, Minneapolis, Minnesota. Association for Computational Linguistics.
- Eleni Metheniti, Tim Van de Cruys, and Nabil Hathout. 2020. [How relevant are selectional preferences for transformer-based language models?](#) In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1266–1278, Barcelona, Spain (Online). International Committee on Computational Linguistics.
- Paolo Pedinotti, Giulia Rambelli, Emmanuele Chersoni, Enrico Santus, Alessandro Lenci, and Philippe Blache. 2021. [Did the cat drink the coffee? challenging transformers with generalized event knowledge.](#) In *Proceedings of \*SEM 2021: The Tenth Joint Conference on Lexical and Computational Semantics*, pages 1–11, Online. Association for Computational Linguistics.
- Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. 2011. [Scikit-learn: Machine learning in python.](#) *Journal of Machine Learning Research*, 12(85):2825–2830.
- Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. [How multilingual is multilingual BERT?](#) In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4996–5001, Florence, Italy. Association for Computational Linguistics.
- James Pustejovsky. 1991. [The Generative Lexicon.](#) *Computational Linguistics*, 17(4):409–441.
- James Pustejovsky. 1995. *The Generative Lexicon*. MIT Press, Cambridge, MA.
- James Pustejovsky and Elisabetta Jezek. 2008. [Semantic coercion in language: Beyond distributional analysis](#), volume 20 of *Italian Journal of Linguistics / Rivista di Linguistica* ∴. De Gruyter Mouton, Berlin, Boston. 2010.
- Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. [Stanza: A python natural language processing toolkit for many human languages.](#) In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 101–108, Online. Association for Computational Linguistics.
- Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. [BERT rediscovers the classical NLP pipeline.](#) In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601, Florence, Italy. Association for Computational Linguistics.
- Laure Vieu, Elisabetta Jezek, and Tim Van de Cruys. 2015. [Quantitative methods for identifying systematic polysemy classes.](#) In *6th Conference on Quantitative Investigations in Theoretical Linguistics (QITL 2015)*, pages 1–5.
- Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. [Transformers: State-of-the-art natural language processing.](#) In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.
- Shijie Wu and Mark Dredze. 2020. [Are all languages created equal in multilingual BERT?](#) In *Proceedings of the 5th Workshop on Representation Learning for NLP*, pages 120–130, Online. Association for Computational Linguistics.
- Mengjie Zhao, Philipp Dufter, Yadollah Yaghoobzadeh, and Hinrich Schütze. 2020. [Quantifying the contextualization of word representations with semantic class probing.](#) In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1219–1234, Online. Association for Computational Linguistics.

## A Models, Parameters and Libraries Used for the Experiments

The embeddings are extracted using the Transformers library (Wolf et al., 2020). The classifiers are trained using the Scikit-learn library (Pedregosa et al., 2011). The Scikit-learn library is also used for the clustering of the negative dataset.

For the SVM algorithm, the radial basis function kernel is used with a C value of 100 and a gamma value of 0.001. K-Means is used as the clustering algorithm for the negative dataset, and the number of clusters ( $k$ ) is determined as 10.

We use the BERT model *dbmdz/bert-base-italian-base-cased* as feature extractor for monolingual Italian classifiers and multilingual LMs *mBERT bert-base-multilingual-cased* and *XLNet xlm-roberta-base* for multilingual classifiers. All models are available at <https://huggingface.co/>.

## B Comparison of Classification Algorithms

The performance of several classification algorithms (*Logistic Regression, Naive Bayes, Random Forest, Support Vector Machine*) is compared for

| Model              |    | Languages |          |          |          |          |          |          |          |          |          |                        |          |          |          |
|--------------------|----|-----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|------------------------|----------|----------|----------|
| Verbal Predication |    |           |          |          |          |          |          |          |          |          |          | Adjectival Predication |          |          |          |
|                    |    | it        |          | de       |          | en       |          | tr       |          | zh       |          | it                     |          | en       |          |
|                    |    | <i>p</i>  | <i>r</i> | <i>p</i> | <i>r</i> | <i>p</i> | <i>r</i> | <i>p</i> | <i>r</i> | <i>p</i> | <i>r</i> | <i>p</i>               | <i>r</i> | <i>p</i> | <i>r</i> |
| Arti.              | B  | 0.94      | 0.95     | -        | -        | -        | -        | -        | -        | -        | -        | 0.94                   | 0.76     | -        | -        |
|                    | mB | 0.93      | 0.92     | 0.90     | 0.79     | 0.92     | 0.93     | 0.91     | 0.64     | 0.92     | 0.76     | 0.91                   | 0.88     | 0.92     | 0.94     |
|                    | XR | 0.90      | 0.91     | 0.88     | 0.83     | 0.94     | 0.93     | 0.93     | 0.84     | 0.92     | 0.92     | 0.93                   | 0.82     | 0.91     | 0.95     |
| Event              | B  | 0.94      | 0.97     | -        | -        | -        | -        | -        | -        | -        | -        | 0.88                   | 0.88     | -        | -        |
|                    | mB | 0.94      | 0.91     | 0.90     | 0.86     | 0.97     | 0.91     | 0.91     | 0.81     | 0.92     | 0.88     | 0.86                   | 0.86     | 0.85     | 0.68     |
|                    | XR | 0.94      | 0.88     | 0.82     | 0.97     | 0.92     | 0.95     | 0.88     | 0.94     | 0.84     | 0.99     | 0.78                   | 0.84     | 0.90     | 0.79     |
| Hum.               | B  |           | 0.93     | 0.92     | -        | -        | -        | -        | -        | -        | -        | -                      | -        | -        | -        |
|                    | mB | 0.92      | 0.92     | 0.96     | 0.90     | 0.98     | 0.98     | 0.97     | 0.74     | 0.96     | 0.83     | -                      | -        | -        | -        |
|                    | XR | 0.93      | 0.96     | 0.98     | 0.98     | 0.98     | 0.98     | 0.96     | 0.95     | 0.97     | 0.91     | -                      | -        | -        | -        |
| Info.              | B  | 1         | 0.97     | -        | -        | -        | -        | -        | -        | -        | -        | 0.86                   | 0.96     | -        | -        |
|                    | mB | 1         | 0.97     | 0.93     | 0.70     | 1        | 0.97     | 0.96     | 0.74     | 1        | 0.82     | 0.86                   | 0.95     | 0.97     | 0.91     |
|                    | XR | 0.97      | 0.97     | 0.97     | 1        | 0.97     | 0.93     | 0.97     | 0.93     | 0.97     | 0.97     | 0.90                   | 0.92     | 1        | 0.80     |
| Loc.               | B  | 0.95      | 0.95     | -        | -        | -        | -        | -        | -        | -        | -        | -                      | -        | -        | -        |
|                    | mB | 0.91      | 0.92     | 1        | 0.85     | 0.92     | 0.97     | 0.97     | 0.56     | 0.93     | 0.92     | -                      | -        | -        | -        |
|                    | XR | 0.97      | 0.94     | 1        | 0.94     | 0.98     | 0.95     | 0.96     | 0.83     | 0.96     | 0.90     | -                      | -        | -        | -        |

Table 3: The test results of each classifier. Precision, Recall scores are given.

|              | Artifact    | Event       | Human       | Info.       | Loc.        |
|--------------|-------------|-------------|-------------|-------------|-------------|
| Log. Reg.    | 0.94        | <b>0.95</b> | <b>0.92</b> | <b>0.98</b> | <b>0.95</b> |
| Naive Bayes  | 0.87        | 0.92        | <b>0.92</b> | 0.97        | 0.92        |
| Rand. Forest | 0.88        | 0.91        | 0.90        | 0.97        | 0.94        |
| SVM          | <b>0.95</b> | <b>0.95</b> | <b>0.92</b> | <b>0.98</b> | <b>0.95</b> |

Table 4: Performance of different classification algorithms on Italian verbal predication test data. Best performing classifiers for each semantic type are given in bold.

the monolingual verbal predication classification task. See Table 4 for the f1 scores of the classifiers trained with different algorithms. Overall, SVM is the best performing one.

## C Data Correction Guideline

Please, follow these points for the manual correction of the translated test data:

- If the verb and the object are not identified correctly, they should be annotated manually.
- The sentences should be corrected if they sound unnatural or the predicate does not target the desired semantic type.
- For the correction, the sentences can be changed or the verb and the noun can be changed.
- The noun should be the object of the verb. If the verb takes a prepositional phrase instead, it should be changed with another verb.
- If any of the target words is a *multi-word expression*, the headword should be considered as the target word.
- If the sentence is passivized in translation, it should be turned into an active one.

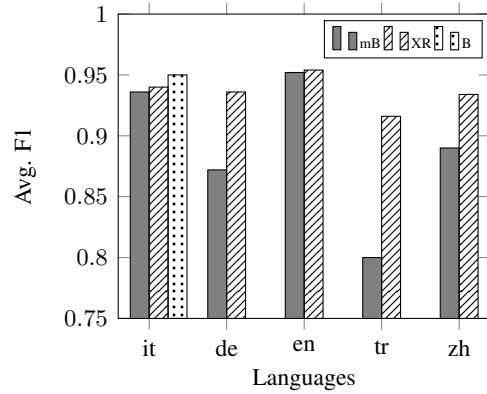


Figure 1: The average f1 score of different model-based classifiers on each language for verbal predication.

## D Detailed Test Results

See Table 3 for the precision and recall scores of each classifier.

## E Language-Specific Evaluation

The performance of the multilingual classifiers for verbal predication changes depending on the target language (See Figure 1). Similar to the studies that investigate the effects of structural differences of languages on cross-lingual transfer (Pires et al., 2019; K et al., 2020; Conneau et al., 2020b; Lauscher et al., 2020; Deshpande et al., 2022), our results show that the performance of the XR-classifiers on the typologically more distant languages Turkish and Chinese is worse. Similarly, the mB-classifiers perform worse on Turkish. We don't think the quality of the translations is the reason since native speakers manually checked the translations of these languages. However, even the worst performance is still good with over 0.8 f1 score.

The classifiers perform best on English test data, which is not the source language. One possible reason is that the Italian test data was not manually corrected, in contrast to the target languages. For this reason, the test data for the source language may contain more noise due to, e.g. parsing errors.

Even though XLM-RoBERTa improves the results for all languages, we see that the improvement changes depending on the language. One possible explanation is the larger size of training data for the XLM-RoBERTa model for these languages compared to mBERT.

## F Food•Event Nouns

*pranzo* ('lunch'), *cena* ('dinner'), *colazione* ('breakfast'), *merenda* ('snack'), *aperitivo* ('aperitif'), *buffet* ('buffet'), *picnic* ('picnic'), *pasto* ('meal'), *spuntino* ('snack')

## G Data Information for Copredication Test Data

The test data contains 30 positive and 24 negative examples of copredication with different semantic types targeting a Food•Event noun. There are both *Food verb-Event adj* and *Event verb-Food adj* combinations in the positives. Similarly, there are both *Food verb-Food adj* and *Event verb-Event adj* combinations in the negatives. The distributions of the types can be seen in Table 4. The cross-lingual copredication test data contains the same number of sentences and distribution for all languages, except for Chinese, which lacks one sentence for *Food verb-Event adj* and one sentence for *Event verb-Event adj*.

| Positives              | Negatives              |
|------------------------|------------------------|
| Total: 30              | Total: 24              |
| <i>food-event</i> : 15 | <i>food-food</i> : 15  |
| <i>event-food</i> : 15 | <i>event-event</i> : 9 |

Table 5: Data size and type distribution of copredication test data. The first semantic type refers to the verbal predication and the second one to the adjectival predication, e.g. *food-event*: Food verb-Event adj.

In addition to these data, another type of negative instances is created in order to test the effects of the complex type nouns in copredication detection. This data contains negative instances of copredication with simple type nouns, in which both a verb and an adjective targeting the same semantic type

(also the same as the noun’s semantic type) predicate over the noun. This type of negative instances are produced by substituting the Food•Event nouns in the negatives with a Food or Event simple type noun. However, in some cases, the sentences are also changed in order to make them more natural. In 24 sentences, 9 sentences are exactly the same except for the noun. However, 10 sentences are changed to some extent, leaving the predicates the same, and 5 sentences are changed completely.

## Chapter 3

### Cococorpus: a corpus of copredication

# Cococorpus: a corpus of copredication

Long Chen   Deniz Ekin Yavaş   Laura Kallmeyer   Rainer Osswald

Heinrich Heine University Düsseldorf

{chen.long, deniz.yavas, laura.kallmeyer, rainer.ossward}@hhu.de

## Abstract

While copredication has been widely investigated as a linguistic phenomenon, there is a notable lack of systematically annotated data to support empirical and quantitative research. This paper gives an overview of the ongoing construction of Cococorpus, a corpus of copredication, describes the annotation methodology and guidelines, and presents preliminary findings from the annotated data. Currently, the corpus contains 1500 gold-standard manual annotations including about 200 sentences with copredications. The annotated data not only supports the empirical validation for existing theories of copredication, but also reveals regularities that may inform theoretical development.

## 1 Introduction

Inherently polysemous nouns have multiple meaning facets. For example, the noun ‘breakfast’ has an object facet referring to its food reading, and an event facet referring to its dining reading. It is common to analyze its semantic type as a “dot-type” *food • event* (Pustejovsky, 1995; Cruse, 1995). Different meaning facets of a dot-type noun can be predicated by multiple predicates at the same time. This phenomenon is referred to as *copredication* (Pustejovsky, 1995; Asher, 2011). In (1), the verb ‘bring’ targets the object facet while ‘late’ targets the event facet.<sup>1</sup>

- (1) Go *bring* your new sister some *late* **breakfast**.

*Coercion*, by contrast, refers to the phenomenon where the predicate targets a facet which is not available in the noun. (2) is an example of coercion.

The verb ‘resist’ targets an event facet, but ‘novel’ is an *object • info* noun without event facets.

- (2) I *resisted* a second **novel** for 14 years until Jack became a way out of a trap I got myself into with a multi-book contract.

Copredication has been studied extensively in linguistics. For example, a number of studies focus on the restrictions and the orders of copredication. Asher (2011) observed an asymmetry in the copredication of the polysemous noun ‘city’. Retoré (2014) noticed that *football team* reading of the polysemous noun ‘Liverpool’ cannot copredicate with other readings. Chatzikyriakidis and Luo (2015) concluded that the copredication of ‘newspaper’ related to the *organization* reading is relatively restricted compared to its other facets. Ortega-Andrés and Vicente (2019) proposed the concept of ‘activation package’ to explain the possibility of the copredication over ‘school’. Sutton (2022) discovered that the *physical entity* and *eventuality* reading of ‘statement’ cannot cooccur in a copredication construction, but either reading can copredicate with the *informational content* reading. Murphy (2024) claimed that complexity and coherence are the decisive factors for the predicate order within a copredication. Michel and Löhr (2024) further suggested that context is a more fundamental factor and explained the order of copredication with the notion of “expectation”. Chen et al. (2025) proposed a distinction between ‘primary facets’ and ‘secondary facets’ to explain the asymmetry in the copredication of *food • event* nouns. However, most of these previous work is based on a small number of introspectively constructed sentences or cherry-picked typical cases, and there is little quantitative research on copredication to prove or disprove the proposed theories. Also, previous work mainly focused on prototypical copredication

<sup>1</sup>The example sentences in the paper are all taken from our annotated data, which come from BookCorpus (Zhu et al., 2015), accessed via <https://huggingface.co/datasets/bookcorpus/>.

instances, while a lot of borderline cases are not well-represented.

These limitations show the need of an annotated corpus of sentences with copredication. Currently there are few corpora related to copredication. Hanks and Pustejovsky (2005) created a lexicon with corpus usage patterns of words with semantic types information. However, the focus is on verbs instead of polysemous nouns and copredication constructions are not specifically addressed. Alonso et al. (2013) annotated in total 4500 sentences from three languages containing regular polysemous nouns, but most of the sentences contain single predication instead of copredications, and copredication is only treated as a kind of underspecification. Another valuable resource focusing on semantic types targeted by predicates is T-PAS (Typed Predicate Argument Structures; Jezek et al. 2014). T-PAS offers argument structure patterns for Italian verbs, annotated with semantic types. It also provides corpus instances for each verb pattern. However, although T-PAS includes a range of semantic types and is not limited to simple type nouns, instances involving dot-types are relatively infrequent, and the focus is on single-type predication. To our knowledge, no corpus to date has a specific focus on copredication.

In this paper we describe the construction of Cocorpus, an ongoing project that aims at a corpus with copredication and coercion annotations. The current version is restricted to English. Up to now, we have mainly been targeting the annotation of copredication, covering three kinds of dot-types, and we have manually annotated about 1500 gold-standard sentences from BookCorpus (Zhu et al., 2015) where a polysemous noun is predicated by multiple predicates, including around 200 copredication sentences. Cocorpus also contains around 18000 silver-standard sentences acquired through automatic annotation.

## 2 Annotation overview

For our copredication annotation, we focused on three common dot-types in language: *food • event*, *info • event*, and *object • info*, which are related to the three major ontological categories *phys(ical)-obj(ect)*, *information*, and *event*.<sup>2</sup> These dot-types are selected because they are relatively better-studied and their facets are rather easy to distinguish from each other.

<sup>2</sup>*food* is a subtype of *phys-obj*

For each dot-type, five nouns with relatively high frequency and little ambiguity are selected. The selected nouns are listed in Table 1. At the moment, we focus on the copredication construction V+Adj+N for annotation. The source of our data is BookCorpus (Zhu et al., 2015), not only because it is free and easily accessible with a considerable number of sentences, but also because of the diversity of the genres of the texts and the contemporary, naturalistic language of the texts.

The annotation pipeline includes the following stages:

1. Automatic extraction of target constructions containing the selected nouns
2. Preliminary annotation using the classifier from Yavas et al. (2023)
3. Manual annotation by two trained annotators
4. Disagreement resolution through discussion

More specifically, the annotation focuses on the relationship between each selected noun and the adjective or verb in the sentence that predicates the noun. For example, in (3), the relation between ‘ate’ and ‘breakfast’ would be annotated as ARTIFACT,<sup>3</sup> and the relation between ‘quick’ and ‘breakfast’ would be annotated as EVENT.

- (3) They packed their bags, *ate* a *quick breakfast* of dry cereal and headed south.

Our labeling scheme primarily follows the original labels of the classifier. The basic facet selection labels consists of ARTIFACT for the predication over the object facet, INFORMATION for the predication over the info(rmation) facet, and EVENT for the predication over the event facet. To account for cases where a predicate simultaneously targets multiple facets,<sup>4</sup> we added four composite labels: ARTIFACT\_INFO, ARTIFACT\_EVENT, EVENT\_INFO, and ARTIFACT\_EVENT\_INFO. Furthermore, coercion is distinguished from facet selection during human annotation, so four additional labels are employed: COERCION\_ARTIFACT, COERCION\_EVENT, COERCION\_INFO and COERCION\_OTHER, which indicate

<sup>3</sup>This label stands for the object facet. It comes from the T-PAS taxonomy, from which our classifier has been trained on. In this paper the annotation labels are presented in small capitals, and semantic types are presented in italics.

<sup>4</sup>e.g. in ‘read a book’, ‘read’ targets both the object facet and the info facet of ‘book’.

| Dot types            | Selected nouns                                      |
|----------------------|-----------------------------------------------------|
| <i>food • event</i>  | breakfast, buffet, dinner, feast, meal              |
| <i>info • event</i>  | conversation, lecture, response, speech, submission |
| <i>object • info</i> | brochure, diary, novel, summary, textbook           |

Table 1: The selected nouns of each dot-type

coercion to the object facet, event facet, info facet, and other facets, respectively. Although theoretically, facet selection and coercion can both happen in one single predication, and coercion can involve multiple facets, our current annotated corpus does not contain such instances, and therefore the corresponding combined labels are not implemented yet. Additionally, it is usually unclear which facets light verbs target in copredication constructions, so we specifically annotate light verbs with the label LVB and exclude them in our current research, such as the relation between ‘have’ and ‘dinner’ in the light verb construction ‘have an early dinner’. And for technical reasons, deleting sentences directly during annotation is not possible, so we also employ a label called DELETE to mark the exclusion of a sentence.

### 3 The construction of the corpus

#### 3.1 Automatic extraction and preliminary classification

We extract sentences containing the candidate nouns from the BookCorpus and parse them using spaCy’s transformer-based pipeline for English.<sup>5</sup> Our goal is to identify sentences where candidate nouns are the direct object of a verb (*dobj*) and modified by an adjective (*amod*) simultaneously. Sentences meeting both criteria are classified in the next step using the classifiers developed by Yavas et al. (2023) for copredication detection.

In their study, Yavas et al. (2023) develop multilingual classifiers to identify semantic argument types in both verbal and adjectival predications. They train separate binary support vector machine classifiers for several semantic types. These classifiers employ contextualized word embeddings generated by pre-trained language models, specifically the multilingual RoBERTa model (Conneau et al., 2020). Given their contextualized word embeddings in the sentences as input, the classifiers classify the relation between the predicate and its argument based on the targeted semantic type. Fig. 1

illustrates the classification process. Yavas et al.

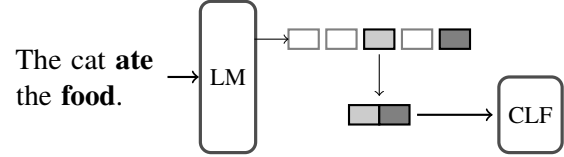


Figure 1: The figure illustrates the working principles of the classifiers developed by Yavas et al. (2023). The classifiers are trained to classify the relation between a predicate and its argument in a sentence using their contextualized word embeddings from a pre-trained language model as input.

(2023) demonstrate that these classifiers effectively detect verb-adjective copredications across multiple languages, making them well-suited for our study. Specifically, we employ six classifiers corresponding to three semantic types for both verbal and adjectival predications: *information*, *event*, and *artifact*. An example of copredication detection using the classifiers is illustrated in Fig. 2.

#### 3.2 Manual annotation

##### 3.2.1 Annotation platform and format

The manual annotation and adjudication were conducted using the INCEpTION annotation platform (Klie et al., 2018).<sup>6</sup> As illustrated in Fig. 3, the annotation interface displays the automatic annotation, with each relation between a predicate and a noun represented by a labeled directional arrow. Annotators could modify the relations by selecting the corresponding arrows and adjusting the assigned labels through a drop-down menu on the right side of the page.

The adjudication interface (Fig. 4) employs a color-coded system to indicate the annotation status of the sentences. On the left side, the green cells and the white cells mark sentences that require no further modification: Green indicates that the annotators did not change the automatic annotation, and white indicates that they changed it in the same way. Red cells indicate unresolved disagreements.

<sup>5</sup>[spacy.io/models/en#en.core.web.trf](https://spacy.io/models/en#en.core.web.trf)

<sup>6</sup>[inception-project.github.io](https://inception-project.github.io)

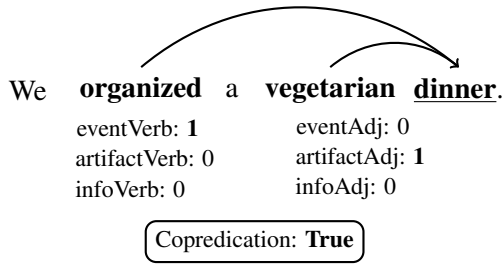


Figure 2: Classification of verbal and adjectival predication in a sentence. Copredication is detected when different types of classifiers assign positive labels to different predication.

The adjudication panel presents a comparative view of both annotators' decisions in the central display area. The adjudicator could either select one of the existing annotations or create a new annotation in the upper panel to establish the final decision.

### 3.2.2 Annotation guidelines

Our annotation distinguishes facet selection from argument coercion. For example, in (4-a), the predicate 'finish' targets the type event, which is not a meaning facet of the noun 'novel', so the relation between 'finish' and 'novel' is annotated as COERCION\_EVENT;<sup>7</sup> in (4-b), although the noun 'novel' does have an info facet, coercion still occurs because according to the context, the content the person 'posted' is the metadata of the novel (e.g. description or advertisement) instead of the actual content, so the relation should be annotated as COERCION\_INFORMATION instead of INFORMATION.

- (4) a. I really have to *finish* this current **novel** before researching another one.  
 b. I have *posted* the new **novel** Rome's Evolution on Amazon, B & N, Kobo and Smashwords.

Regarding facet selection, the nouns in the current annotation task are preselected, so it is clear which facets these nouns have. The major annotation task thus reduces to identifying which facets are targeted by given predicates. Operationally, we distinguish selected facets by substitution. Taking the object facet as an example, replace the dot-type noun with physical objects such as 'stone'. The

<sup>7</sup>While events such as reading and writing are often analyzed as being part of the qualia structure of 'novel', they do not count as formal quale but as telic quale (cf. Pustejovsky 1995, Sect. 6.2); that is, they do not qualify as event facets of the noun.

phrase 'throw the stone' is felicitous but '#memorize the stone' is semantically anomalous, which proves that 'throw' targets object facets but 'memorize' cannot.

The facet selection of certain predicates is context-dependent, requiring annotators to determine the most possible targeted facets based on the context from the sentence. As a representative case, the verb 'remember' can target only the object facet of 'breakfast' (as in (5-a)) or target both the event and the object facets of 'meal' simultaneously (as in (5-b)).

- (5) a. Her pleased expression tells me she likes that I *remember* her favorite **breakfast**.  
 b. Once passed the initial security checkpoint it occurred to Jane that she could not *re-member* her last **meal**, but there were lines outside all the food stalls.

Some predicates exhibit a high flexibility in meaning facet selection or their predication involve other mechanisms instead of standard facet selection. For example, in 'love the book', any part of the book can be targeted by 'love'; in 'his own book', the adjective 'own' is more focused on the possession relation instead of the facets of 'book'. In such cases, determining the targeted facets becomes both methodologically challenging and theoretically uninformative. Therefore, these predicates were systematically excluded from our annotation. These predicates include:

- 'like' verbs: like, love, hate, prefer
- 'equal' verbs: be, equal, mean
- quoting verbs: say, mention
- adjectives describing type/token: certain, particular, single, same, such
- adjectives describing number: many, much, more, enough, some, few, extra
- adjectives describing entirety/identity: own, whole, entire, complete, actual, real, other, another
- adjectives describing quality: good, nice, fine, best, great, fantastic, wonderful, amazing, bad, awful

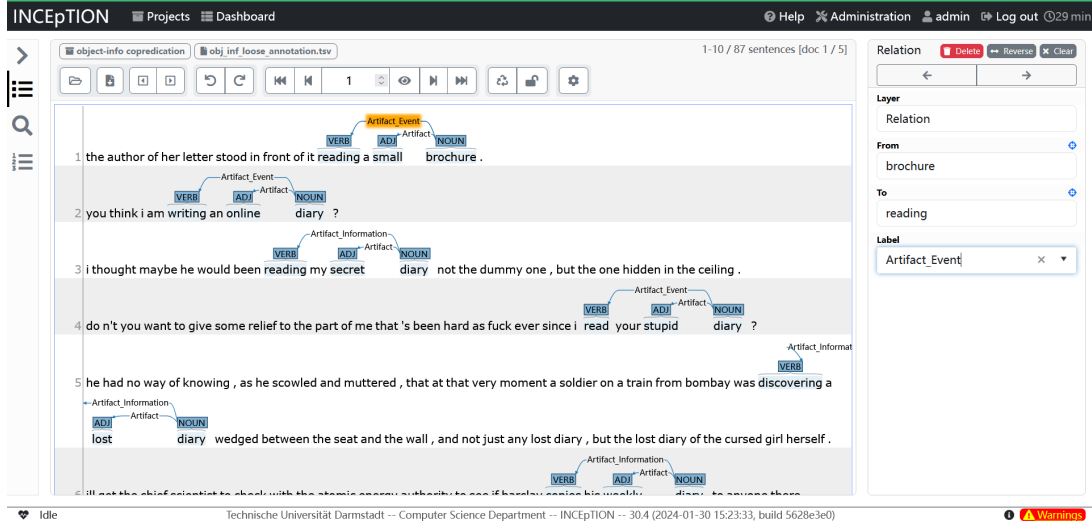


Figure 3: Annotation interface

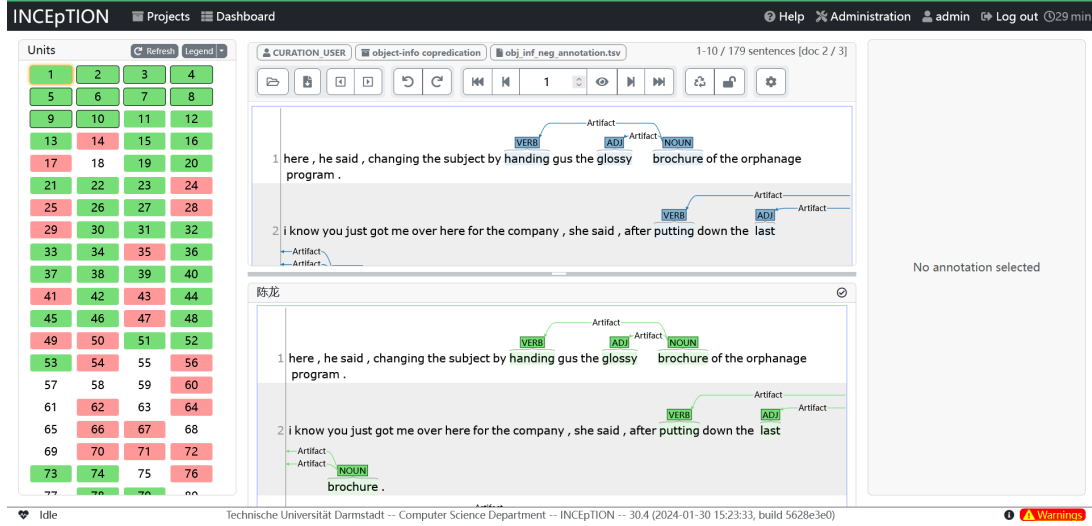


Figure 4: Adjudication interface

## 4 The analysis of the annotation

### 4.1 Statistics of the annotated data

The extraction process yielded varying numbers of candidate sentences across different dot-types. For the selected *food • event* nouns, we extracted 6838 sentences with multiple predication from BookCorpus. From this subcorpus, we randomly selected 300 ‘positive candidates’ where the classifier detected copredication (the verb-noun relation and the adjective-noun relation are different) and 300 ‘negative candidates’ (both predicates target the same facet). For *info • event* nouns, we got 9129 sentences with multiple predication, from which we similarly selected 300 positive and 300 negative cases for annotation. *object • info* nouns provided only 832 sentences in total due to the relatively low

|            | $V_{obj}$ | $V_{ev}$ | $V_{both}$ | $V_{coercion}$ | $\Sigma$ |
|------------|-----------|----------|------------|----------------|----------|
| $A_{obj}$  | 303       | 16       | 12         | 1              | 332      |
| $A_{ev}$   | 108       | 42       | 17         | 2              | 169      |
| $A_{both}$ | 2         | 4        |            |                | 6        |
| $\Sigma$   | 413       | 62       | 29         | 3              | 507      |

Table 2: Statistics of the annotation of *food • event* nouns

frequency of the five chosen nouns. Consequently, we selected only 150 ‘positive’ and 150 ‘negative’ candidates for annotation.

#### 4.1.1 *food • event*

Among the 600 chosen sentences of the five *food • event* nouns, 507 valid annotations were obtained. Around 100 sentences were excluded,

mainly because they involve light verb constructions or parsing errors. Table 2 reveals the frequency of the predicate tokens that target different facets of the nouns.<sup>8</sup> The majority of verbs (413 out of 507) target the object facet, and approximately two-thirds of adjectives (332 out of 507) target the object facet. Additionally, around 30 sentences contain predicates that simultaneously target both facets. Furthermore, three sentences were identified as involving a coercion.

- (6) a. “I *promised* you guys a hot **meal**”, said Ellen, sighing.  
 b. Her expression changes to that of a lioness *stalking* her next **meal**.

(6) presents two annotated instances of coercion. In (6-a), the verb ‘promise’ typically targets an event facet and the sentence means ‘I promised to get you guys a hot meal’. Although the noun ‘meal’ has an event facet, the type of the event facet is *eating*, which is not the giving event ‘promise’ targets. Thus, the relation between ‘promise’ and ‘meal’ is annotated as COERCION\_EVENT. In (6-b), the object of the verb ‘stalk’ usually needs to be an animate object rather than food, so the predication is annotated as COERCION\_OTHER.

While our annotation can provide empirical evidence for some theoretical analyses on copredication, we are not yet able, due to the limited number of dot types and copredication instances, to fully validate or falsify theories related to dot objects with multiple facets (as in Asher, 2011, Ortega-Andrés and Vicente, 2019, Sutton, 2022, etc.) or specific to word items and context (as in Michel and Löhr, 2024). Murphy (2024) proposed a principle called Incremental Semantic Complexity (ISC) and concluded that in copredication, the concrete readings would come earlier than abstract readings (in linear order). It is assumed that physical objects are more concrete than information, and information is more concrete than events. This assumption is supported to a large extent by our annotation statistics on copredication over *food • event* nouns. As shown in Table 2, in most copredication instances, the verb, which precedes the adjective, is targeting the object facet. However, there are still 16 cases where the verb targets the event facet, indi-

cating that the ISC is rather a tendency than a strict principle.

As observed in Chen et al. (2025), for *food • event* nouns in the copredication pattern V+Adj+N, the verb targeting the event facet and the adjective targeting the object facet is a preferred order. When the adjective instead targets the event facet, copredication is only possible when the adjective is *facet-addressing*. Facet-addressing adjectives, contrary to *facet-picking* adjectives, are adjectives that do not affect the availability of a facet of a dot-type noun. For example, ‘quick’ is a facet-addressing adjective, because ‘quick lunch’ is still a dot-type and can be copredicated by object-targeting verbs like ‘cook’ and ‘order’. By contrast, ‘slow’ is a facet-picking adjective since the object facet is not available anymore in ‘slow lunch’, and ‘#cook a slow lunch’ or ‘#order a slow lunch’ are not acceptable. However, according to Table 2, only 16 cases align with the assumption that object facet is targeted first, whereas there are 108 cases where the copredication works in the less preferred direction. This discrepancy can be partially explained by the dominance of object-targeting verbs in the corpus.

Further analysis of the 108 copredication cases with the event-targeting adjectives reveals that there are only 23 adjective types in the 108 cases. These adjectives fall into the following four semantic categories:

- order related: *first, last, next, fourth, new*
- time related: *quick, slow, early, late, occasional*
- specialness related: *special, customary, regular, unexpected, obligatory, worthy, easy*
- other: *romantic, solitary, civilized, corporate*

Notably, all of them except ‘slow’ are facet-addressing adjectives. In all the copredication instances involving ‘slow’, the verb was always ‘eat’ in our dataset. We discovered in previous studies that ‘eat’ can also take *food • event* nouns modified by any adjectives as objects, probably due to its relatively light meaning in the context of meals. Moreover, the phrase ‘eat a slow meal’ could be understood as eating a meal slowly, which makes the phrase felicitous.

<sup>8</sup> $V_{obj}$  stands for the cases where the verb targets the object facet;  $V_{both}$  means that the verb targets both facets;  $V_{coercion}$  means the verb is annotated as having coercion. Similar interpretations apply to the other symbols of the table.

|           | $V_{inf}$ | $V_{ev}$ | $V_{both}$ | $V_{coercion}$ | $\Sigma$ |
|-----------|-----------|----------|------------|----------------|----------|
| $A_{inf}$ | 53        | 28       | 1          | 1              | 83       |
| $A_{ev}$  | 14        | 189      | 2          |                | 205      |
| $\Sigma$  | 67        | 117      | 3          | 1              | 288      |

Table 3: Statistics of the annotation of *info • event* nouns

#### 4.1.2 *info • event*

From the initial set of 600 chosen sentences containing *info • event* nouns, only 288 yielded valid annotations. The low proportion of valid annotations is due to two factors: (1) the selected nouns are deverbal nouns and frequently occur in light-verb constructions, and (2) ‘response’, ‘submission’ and ‘speech’ also have other meanings unrelated to info facets. The distribution of facet selection for these *info • event* nouns is presented in Table 3.

In the 288 instances of multiple predication, there are only 42 instances (14.6%) of copredication. In 14 of the instances, the verb targets the info facet and the adjective targets the event facet; in the other 28 instances copredication works in the other order. This implies either the ISC from [Murphy \(2024\)](#) might be too strict or *information* and *event* are actually close to each other in terms of complexity.

The proportion of copredication over *info • event* nouns is significantly lower than that observed with *food • event* nouns. This is consistent with the observation in [Chen et al. \(2025\)](#) that for *info • event* nouns, both facets tend to be secondary facets, that might be inaccessible if the other facets are targeted first, and copredication in the construction V+Adj+N can only happen when the adjective is facet-addressing.

The 42 copredication instances only include 10 different adjective types. These adjectives can be classified into the following four semantic categories and they are all facet-addressing predicates:

- order related: *last*
- time related: *rapid, earlier, lengthy, little*
- atmosphere related: *bickering, heated*
- speaker related: *private, unwilling, hasty*

#### 4.1.3 *object • info*

The distribution of predications over *object • info* nouns is presented in Table 4, derived from 253 valid annotations out of 300 candidate sentences.

|                | $V_{obj}$ | $V_{inf}$ | $V_{both}$ | $V_{coercion}$ | $\Sigma$ |
|----------------|-----------|-----------|------------|----------------|----------|
| $A_{obj}$      | 56        | 2         | 12         |                | 70       |
| $A_{inf}$      | 30        | 31        | 96         | 23             | 180      |
| $A_{coercion}$ | 1         | 1         | 1          |                | 3        |
| $\Sigma$       | 87        | 34        | 109        | 23             | 253      |

Table 4: Statistics of the annotation of *object • info* nouns

|                | $V_{obj}$ | $V_{inf}/V_{both}$ | $V_{coercion}$ | $\Sigma$ |
|----------------|-----------|--------------------|----------------|----------|
| $A_{obj}$      | 56        | 14                 |                | 70       |
| $A_{inf}$      | 30        | 127                | 23             | 180      |
| $A_{coercion}$ | 1         | 2                  |                | 3        |
| $\Sigma$       | 87        | 143                | 23             | 253      |

Table 5: Updated statistics of the annotation of *object • info* nouns

This dot-type exhibits a notably higher frequency of coercion, with the coercion to the event facet being particularly predominant. In 22 of the 23 coercion instances, the verb targets an event facet, suggesting a possible tendency of the direction of coercion.

Regarding copredication, the 32 instances revealed a significant asymmetry, which is consistent with the ISC suggested by [Murphy \(2024\)](#) but seems contradictory to the observation in [Chen et al. \(2025\)](#). According to [Chen et al. \(2025\)](#), there is little restriction on the copredication of *object • info* nouns and copredication can happen in both orders. However, in only two cases in our annotated data, the verb targets the event facet and the adjective targets the object facet. This may be attributed to two factors. First, the frequency of info-targeting adjectives is relatively high (180 out of 253 instances). Secondly, high-frequency verbs like ‘read’, ‘write’, and ‘publish’ are annotated as targeting both facets, resulting in a high number in the third column of the table. Interestingly, in 96 of the 109 instances, the adjective targets the info facet, suggesting that verbs like ‘read’ probably “mainly” targets the info facet. If we take this into account and combine the cases where the verb targets the info facet and the verb targets both facets, the updated statistics (as in Table 5) reveals a more balanced distribution of copredications.

## 4.2 Disagreement analysis

The inter-annotator agreement is listed in Table 6. The primary reasons of inter-annotator disagree-

| Dot types           | Agreement |
|---------------------|-----------|
| <i>food • event</i> | 0.64      |
| <i>info • event</i> | 0.43      |
| <i>obj • info</i>   | 0.67      |

Table 6: The inter-annotator agreement in Cohen’s Kappa

ment can be summarized as follows.

#### 4.2.1 The exclusion of the sentences

As is in (7), the adjective ‘complete’ refers to part-whole relations and can target any facets; so one of the annotator followed the guideline and labeled DELETE while the other annotator decided by context, which suggests that ‘complete’ targets the info facet, and annotated INFORMATION.

- (7) To buy the *complete* **novel**, Trail of Storms, [click here](#).

#### 4.2.2 Difficult contexts

In some sentences, the predicate can target both facets of the noun, but the context provided by the sentence is insufficient or complicated, which also results in the disagreement between annotators.

- (8) a. Do you *remember* that first **dinner**?  
b. He was cooking a *special* **dinner** for her and he had finally found the perfect ring to put on her finger, a heart shaped diamond surrounded by smaller stones and set in platinum.

The annotation of (8-a) presented a challenge due to lack of context. The annotators labeled ARTIFACT and EVENT respectively. The adjudication process selected EVENT as the final decision, because conceptually, it is more plausible to recall a dining event while forgetting specific culinary details than remembering only the food while forgetting the associating eating event. Thus, the event facet is established as the default selection for such contextually underspecified cases in terms of remembering a meal.

In (8-b), the context provides competing clues. The verb ‘cook’ implies a special food preparation, while the sentential context subsequently indicates the dining experience being ‘special’, resulting in the divergent annotation of the relation between ‘special’ and ‘dinner’. The final decision is that

‘special’ targets the event facet, as the contextual evidence provided no substantive indication of unusual food characteristics that warrants an object facet selection. This annotation example reflects the difficulty in the identification of copredication regarding predicates with a wider choice of facets.

#### 4.2.3 Borderline light verbs

There is little consensus regarding the definition of light verb constructions, which is reflected in our annotation of high-frequency *info • event* noun patterns including ‘give a speech/lecture’, ‘make a response/speech/conversation’ and ‘deliver a speech’. One annotator label them as LVB and the other treat them as regular verb phrases, contributing substantially to the relatively low agreement in the annotation of *info • event* nouns.

To resolve the disagreement, we implemented the diagnostics proposed by Fleischhauer and Neisani (2020), such as replacing the verb with its synonyms and examining the acceptability and the meaning of the new phrase. Application of these diagnostics reveal that the verbs ‘deliver’, ‘give’ and ‘make’ cannot be easily substituted (‘#send a speech/lecture’, ‘#produce a conversation’ are ungrammatical). Consequently, the verbs mentioned above are regarded as light verbs during the final adjudication.

#### 4.2.4 Unclear distinctions between facets

Some of the disagreement arises from the unclear distinctions between facets, especially the object facet and the info facet of *object • info* nouns.

- (9) a. He tried to talk a lot about theories and make funny stories at times to let students feel like they were not drones *downloading* the latest **textbook** that the publishing company decided could be revised for the twelfth time in a row for twelve years straight.  
b. During this time, she published a *short* **novel**.

The ontological status of digital texts presents a significant challenge to our annotation, as illustrated in example (9-a), which involves the predication over electronic versions of textbooks rather than traditional physical books. It is arguable whether the PDF file and the strings in the computers are a kind of physical object or more about the information.

A similar puzzle also exists with traditional paper books, as demonstrated in example (9-b). The

adjective ‘short’ unambiguously targets the info facet of the ‘novel’, but at the same time, the length of the printed characters in the physical book is also short. The printed symbols are ontologically different from the object facet of the ‘novel’, which is usually made of paper and consists of covers, and also different from the info facet of the ‘novel’, which does not have a physical form. Frequent verbs including ‘read’ and ‘write’ also have the same problem. The entity a person ‘read’ in a novel is not the paper material but rather the printed matter on the paper, which is not exactly the object facet of ‘novel’. The statistics shown in Fig. 4 also suggests that the facet these verbs target is probably closer to info facet than object facet. Currently, the label INFORMATION is decided for both controversial cases, but these disagreements highlight a need for a comprehensive revisit of the analysis of *object • info* nouns and an investigation of the possible existence of a third meaning facet.

#### 4.2.5 Borderline coercions

The unclear distinction between coercion and facet selection is also a reason for the disagreement between annotators.

- (10) a. Anticipating an *angry* **conversation** he would not want to overhear, Mark hurried to the shower.  
 b. It will handle all your daily chores, provide *intelligent* **conversation** and need absolutely no maintenance.

The examples (10-a) and (10-b) exemplify a systematic pattern of human-targeting adjectives modifying *info • event* nouns, relating to the behavioral manner of the participant during the event. On the one hand, the constructions display typical features of coercion. For example, the syntactic transformation of these phrases are restricted. Transformations such as ‘?The conversation is angry’, ‘#It provides a conversation that is intelligent’ are marginally acceptable or unacceptable. Furthermore, these adjectives cannot easily modify the event facet or info facet of the nouns of other types, e.g. ‘?an angry book’ requires some context to be acceptable, and ‘#an intelligent meeting’ is infelicitous.

On the other hand, the treatment of these cases as facet selection can also be justified. First, the interpretation of the phrase is specific, unlike the typical coercion examples such as ‘finish the book’,

where the event that is ‘finished’ is implicit and needs to be specified by context. Second, the usage of human-targeting adjectives for *info • event* nouns is productive. Other adjectives of this kind including ‘cheerful’, ‘friendly’, ‘polite’, and ‘honest’ modifying *info • event* nouns also exist in our annotated data.

Currently, these instances are annotated as facet selection. A more comprehensive analysis and annotation on coercion will be left for future research.

## 5 Conclusion and future work

The construction of Cococorpus is an ongoing project. Currently, we achieved an annotation of more than 1000 sentences with multiple predication over inherently polysemous nouns, among which 198 sentences exhibit copredication. The annotated data can serve as an empirical evidence for some linguistic analyses on copredication, such as a tendency of ISC from Murphy (2024) and the distinction between primary and secondary facets of polysemous nouns from Chen et al. (2025). The annotated data and annotation guideline are published at the project Github page (<https://github.com/CoCoCo-Project>).

Many aspects of copredication remain to be addressed in future annotation efforts. The current annotation framework is limited to the construction V+Adj+N, while other typical copredication constructions, such as (reduced) relative clauses and multiple adjectives, are yet to be incorporated. Furthermore, the existing coverage of dot-types and nouns is also relatively limited. We plan to expand our annotation scope to include: (1) additional dot-types and lexical items, particularly nouns exhibiting multiple facets (e.g., *school*, *city*) and those with debatable facet classifications (e.g., *annotation*); (2) a broader range of predicate types; (3) cross-linguistic investigations to examine potential variations in copredication phenomena across different languages.

## Acknowledgements

This work is part of the project “Coercion and Copredication as Flexible Frame Composition” funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation), project number 440934416. We would like to thank the anonymous reviewers for their valuable comments.

## References

- Héctor Martínez Alonso, Bolette Sandford Pedersen, and Núria Bel. 2013. Annotation of regular polysemy and underspecification. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 725–730.
- Nicholas Asher. 2011. *Lexical meaning in context: A web of words*. Cambridge University Press.
- Stergios Chatzikyriakidis and Zhaohui Luo. 2015. *Individuation criteria, dot-types and copredication: A view from modern type theories*. In *Proceedings of the 14th Meeting on the Mathematics of Language (MoL 2015)*, pages 39–50.
- Long Chen, Laura Kallmeyer, and Rainer Osswald. 2025. Primary vs. secondary meaning facets of polysemous nouns. *Empirical issues in syntax and semantics: Selected papers from CSSP 2023*, page 27.
- Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. *Unsupervised cross-lingual representation learning at scale*. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.
- D. Alan Cruse. 1995. Polysemy and related phenomena from a cognitive linguistic viewpoint. In Patrick Saint-Dizier and Evelyn Viegas, editors, *Computational lexical semantics*, pages 33–49. Cambridge University Press.
- Jens Fleischhauer and Mozghan Neisani. 2020. Adverbial and attributive modification of persian separable light verb constructions. *Journal of Linguistics*, 56(1):45–85.
- Patrick Hanks and James Pustejovsky. 2005. A pattern dictionary for natural language processing. *Revue Française de linguistique appliquée*, 10(2):63–82.
- Elisabetta Jezeq, Bernardo Magnini, Anna Feltracco, Alessia Bianchini, and Octavian Popescu. 2014. *T-PAS; a resource of typed predicate argument structures for linguistic analysis and semantic processing*. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC’14)*, pages 890–895, Reykjavik, Iceland. European Language Resources Association (ELRA).
- Jan-Christoph Klie, Michael Bugert, Beto Boullosa, Richard Eckart De Castilho, and Iryna Gurevych. 2018. The inception platform: Machine-assisted and knowledge-oriented interactive annotation. In *Proceedings of the 27th international conference on computational linguistics: system demonstrations*, pages 5–9.
- Christian Michel and Guido Löhr. 2024. *A cognitive psychological model of linguistic intuitions: Polysemy and predicate order effects in copredication sentences*. *Lingua*, 301:103694.
- Elliot Murphy. 2024. *Predicate order and coherence in copredication*. *Inquiry*, 67(6):1744–1780.
- Marina Ortega-Andrés and Agustín Vicente. 2019. *Polysemy and co-predication*. *Glossa: a journal of general linguistics*, 4(1).
- James Pustejovsky. 1995. *The Generative Lexicon*. MIT Press, Cambridge, MA.
- Christian Retoré. 2014. The Montagovian Generative Lexicon  $\Lambda T y_n$ : a type theoretical framework for natural language semantics. In *TYPES: International Workshop on Types and Proofs for Programs, April 2013, Toulouse, France*, pages 202–229.
- Peter Roger Sutton. 2022. *Restrictions on copredication: a situation theoretic approach*. In *Proceedings of the 32nd Semantics and Linguistic Theory Conference*, pages 335–355.
- Deniz Ekin Yavas, Laura Kallmeyer, Rainer Osswald, Elisabetta Jezeq, Marta Ricchiardi, and Long Chen. 2023. Identifying semantic argument types in predication and copredication contexts: A zero-shot cross-lingual approach. In *Proceedings of the 14th International Conference on Recent Advances in Natural Language Processing*, pages 310–320.
- Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the IEEE international conference on computer vision*, pages 19–27.



## Chapter 4

# Assessing the Significance of Encoded Information in Contextualized Representations to Word Sense Disambiguation

# Assessing the Significance of Encoded Information in Contextualized Representations to Word Sense Disambiguation

Deniz Ekin Yavas

Heinrich Heine University Düsseldorf

deniz.yavas@hhu.de

## Abstract

The similarity of representations is crucial for WSD. However, a lot of information is encoded in the contextualized representations, and it is not clear which sentence context features drive this similarity and whether these features are significant to WSD. In this study, we address these questions. First, we identify the sentence context features that are responsible for the similarity of the contextualized representations of different occurrences of words. For this purpose, we conduct an explainability experiment and identify the sentence context features that lead to the formation of the clusters in word sense clustering with CWEs. Then, we provide a qualitative evaluation for assessing the significance of these features to WSD. Our results show that features that lack significance to WSD determine the similarity of the representations even when different senses of a word occur in highly diverse contexts and sentence context provides clear clues for different senses.

## 1 Introduction

Contextualization is a powerful tool as it enables us to capture sentence context. This is crucial especially in word sense disambiguation (WSD) because sentence context provides valuable information for resolving lexical ambiguity in both NLP and human language processing.

The similarity of representations is crucial for WSD. With contextualization, we expect the representations of different occurrences of the same sense to be similar to each other. This is based on the assumption that different senses of a word occur in different contexts and sentence context contains explicit clues that signal one of the senses of the word. Consider the sentences in (1) that demonstrate two senses of ‘bank’. In both sentences, some words successfully signal each sense of the word; in (1-a), the words ‘money’ and ‘withdraw’ and in (1-b), the words ‘picnic’ and ‘river’.

- (1) ‘bank’ (homonymy):
  - a. *financial institution*:  
I went to the **bank** to withdraw money.
  - b. *geographical feature*:  
They had a picnic by the river **bank**.
- (2) ‘pass’ (polysemy):
  - a. *go across or through*:  
She **passed** through towns.
  - b. *move past*:  
She **passed** the bakery on her way.

However, in practice, we lack clarity on which specific sentence context features are responsible for the similarity of the contextualized representations. It has been shown that a wide variety of information is encoded in the contextualized representations (Sajjad et al., 2022) and using contextualized word embeddings (CWEs) of pre-trained language models alone does not achieve good performance in unsupervised settings (Yenicelek et al., 2020).

The purpose of this study is to investigate which sentence context features determine the similarity of the representations of different occurrences of words and whether these features are significant to WSD. By doing so, we aim to provide a clearer understanding of contextualized representations in terms of their ability to capture different meanings of words. For this purpose, we conduct an explainability experiment. We focus on word sense clustering with CWEs of BERT (Devlin et al., 2019) and identify sentence context features that lead to the formation of the clusters. This way, we determine which features drive the similarity of the representations.

Our cluster explainability method follows several steps and is depicted in Figure 1. We start by performing word sense clustering with CWEs and cluster the sentences of a word. Our aim is essentially to reverse the word sense clustering process

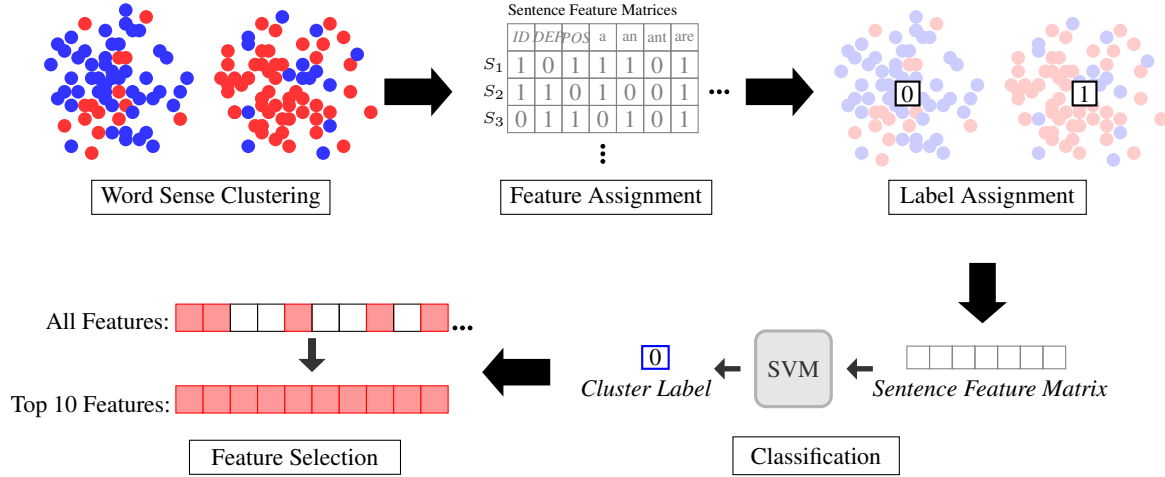


Figure 1: Cluster Explainability Method: i) Perform Word Sense Clustering with CWEs, ii) Represent each sentence with sentence context features, iii) Assign each sentence a label based on the cluster it belongs to, iv) Train a classifier to predict the clusters that sentences belong to based on their sentence context features, v) Apply feature selection to determine the sentence context features responsible for the performance of the classifier.

and recreate the clusters with a classification task. For this purpose, we represent each sentence with its sentence context features and assign it a label based on the cluster that it belongs to. Then, we train a classifier to predict the cluster labels of the sentences based on their sentence context features. As the final step, we apply a feature selection algorithm to the classifier to determine which features are the most relevant for the performance of the classifier. This tells us which sentence context features lead to the formation of the clusters. We use this method to identify the features that lead to the clusters for each word in our dataset. Finally, we assess the significance of these features to WSD for each word through qualitative evaluation.

In this study, we distinguish two types of lexical ambiguity; homonymy and polysemy. The most distinctive feature that distinguishes these two types is the semantic relatedness of their senses (Klepousniotou, 2002; Klepousniotou and Baum, 2007; Klepousniotou et al., 2008, 2012). Homonyms have less semantically related senses compared to polysemes. As a result, homonyms occur in more diverse contexts. Consider the examples in (1) and (2). The provided senses of ‘bank’ are homonymous, whereas those of ‘pass’ are polysemous. The senses of ‘bank’ are not related semantically and the noun co-occurs with semantically different words in its different senses (‘money’, ‘withdraw’ vs. ‘river’, ‘picnic’). However, this is not true for ‘pass’. The meaning difference between the senses of ‘pass’ is less clear and the verb

co-occurs with words that are similar in meaning in its different senses, specifically words that are related to a location.

Their inherent differences also result in differences in NLP performance. For example, WSD performance is better with homonyms (Nair et al., 2020; Haber and Poesio, 2021) and contextualization affects homonyms more (Sevastjanova et al., 2021) compared to polysemes. Therefore, it’s important to consider that not all words present an equal challenge for WSD. In addition to that, it’s important to consider that different lexical ambiguity types have different relations to context, and as a result, the information that is required for their disambiguation might not always be the same. In this study, we expect different results for each type. Considering that homonyms occur in more diverse contexts, we expect sentence context to provide clearer clues for their different senses and the similarity of the representations to be affected by these clues.

Our results show that the sentence context features that are responsible for the similarity of the representations and lead to the formation of clusters lack significance to WSD in most cases. This is true for both lexical ambiguity types. Even with homonyms—where different senses of a word occur in highly diverse contexts and sentence context provides clear clues for different senses—the similarity of the representations does not arise from the significant features.

## 2 Related Work

Studies have shown that the similarity of the embeddings is primarily influenced by the sentence context rather than the meaning of words. [Ethayarajh \(2019\)](#) have shown that words have different representations based on their contextual variation, rather than their meaning variation. Similarly, [Garcia \(2021\)](#) have shown that the similarity between a word and its synonym is lost when the sentence context is identical for different words; the similarity between a word and a random word is not different from the similarity between a word and its synonym when their sentence contexts are similar.

One reason for this is that a lot of information is encoded in the contextualized representations and they affect the similarity of the representations. [Sajjad et al. \(2022\)](#) have shown that the information encoded in the contextualized representations can be explained to some extent by semantic, morphological, syntactic, and lexical concepts. These concepts include the words' POS tags, CCG super-tags, ngrams, casings, WordNet concepts, and so on. Additionally, [Mickus et al. \(2020\)](#) have shown that the similarity of the representations is affected by the segment embeddings that the model assigns to tokens to indicate their sentences. In this study, this is not an issue because we only use one sentence as an input. However, we aim to investigate whether a similar effect can be found for the positional encoding, resulting words in the same position having similar representations.

Clustering reveals these similarities within the representations. [Sajjad et al. \(2022\)](#) have shown that the clusters of contextualized representations overlap with the concepts that are found to be encoded in the representations. Furthermore, it has been shown that word sense clustering with CWEs of the BERT model doesn't achieve good performance and sentence context similarities have been observed within the clusters ([Yenicelik et al., 2020](#)). The effects of the sentence context have been also observed in similarity ranking for WSD with CWEs of BERT ([Gessler and Schneider, 2021](#)). However, there hasn't been any effort to systematically explain the relation between sentence context and contextualized representations of different occurrences of the same word and with a focus on WSD. This study aims to fill this gap.

Finally, the studies that distinguish different types of lexical ambiguity have shown that WSD performance with CWEs changes depending on the

type and it is easier to disambiguate homonymy than polysemy ([Nair et al., 2020](#); [Haber and Poesio, 2021](#)). Similarly, contextualization of BERT affects different types differently and homonyms are affected more by contextualization due to the fact that they occur in more diverse contexts ([Sevastjanova et al., 2021](#)). In this study, we also expect the results to be different for each type. In the case of homonymy, we expect sentence context to provide clearer clues for different senses and the similarity of the representations to be affected by these clues.

## 3 Data

We use SemCor ([Miller et al., 1993](#)) which provides sentences that are annotated with WordNet senses for a wide variety of words ([Fellbaum, 2010](#)) for English. We restrict our focus to nouns and verbs. A word can be both homonymous and polysemous at the same time because different senses of a word can have different relations, e.g. two senses can be homonymous while another two can be polysemous. Because of this, we don't focus on homonymous or polysemous words but sense groups of words. These sense groups are formed by grouping the senses of a word according to their relations, so we end up with sense groups in which all pairs are homonymous or polysemous to each other.

In order to decide if a sense pair is homonymous or polysemous, we use the data provided in [Nair et al. \(2020\)](#). This data contains semantic relatedness judgment scores for a subpart of WordNet. Semantic relatedness determines where on the homonymy-polysemy continuum a word is ([Klepousniotou, 2002](#); [Klepousniotou and Baum, 2007](#); [Klepousniotou et al., 2008, 2012](#)) and semantic relatedness judgments of speakers overlap with different types of lexical ambiguity ([Klepousniotou et al., 2008](#); [Nair et al., 2020](#)). In [Nair et al. \(2020\)](#), semantic relatedness judgement scores are collected for each sense pair from several speakers. We use the average of the scores for each sense pair to decide if the word is homonymous or polysemous in those senses. We consider the sense pairs that have a distance over 0.8 as homonymy pairs and a distance below 0.5 as polysemy pairs.<sup>1</sup>

<sup>1</sup>Psycholinguistics studies have shown that some polysemy types show homonymy-like behaviors and have less semantic relatedness ([Klepousniotou and Baum, 2007](#); [Klepousniotou et al., 2008](#)). Due to this, we leave a certain range out to avoid these mixed types.

| Lexical Ambiguity | Clustering    |      | Sense Group # | Sentence-Feature Classifiers |      |      |             |      |      |             |      |      |          |
|-------------------|---------------|------|---------------|------------------------------|------|------|-------------|------|------|-------------|------|------|----------|
|                   | Sense Group # | ARI  |               | WSD                          |      |      | Cluster     |      |      | 10-Cluster  |      |      | 10-Rand. |
|                   |               |      |               | F1                           | P    | R    | F1          | P    | R    | F1          | P    | R    | F1       |
| Homo.             | 19            | 0.68 | 5             | 0.80                         | 0.80 | 0.80 | <b>0.87</b> | 0.87 | 0.87 | 0.85        | 0.86 | 0.85 | 0.42     |
| Poly.             | 39            | 0.36 | 5             | 0.72                         | 0.76 | 0.72 | 0.78        | 0.80 | 0.78 | <b>0.86</b> | 0.86 | 0.87 | 0.37     |
| Overall           | 58            | 0.52 | 10            | 0.76                         | 0.78 | 0.76 | 0.82        | 0.83 | 0.82 | <b>0.85</b> | 0.86 | 0.86 | 0.39     |

Table 1: Data size and experimental results summary. *WSD* refers to the classifiers that are trained for WSD, *Cluster* for cluster assignment. *10-Cluster* classifiers refer to the classifiers that are trained for cluster assignment with only the top 10 features. *10-Rand.* refers to the classifiers that are trained for cluster assignment with random 10 features. *F1*, *Precision* and *Recall* scores are given. The best F1 score overall and for each type is given in bold.

As our data, we use the sentences of the selected senses from SemCor. We do not include the senses that have less than 10 sentences in SemCor. We balance the number of sentences for each group by random under-sampling. We exclude the words that show inherent and metonymical polysemy because different types of polysemy have different characteristics (Klepousniotou and Baum, 2007; Klepousniotou et al., 2008) and we focus only on irregular polysemy.<sup>2</sup>

## 4 Method

The goal of this study is to first, identify the sentence context features that determine the similarity of the contextualized representations. For this purpose, we identify the sentence context features that lead to the formation of clusters in word sense clustering with CWEs. Then, we evaluate these features’ significance to WSD.

In order to identify these features, we conduct a cluster explainability experiment. First, we perform word sense clustering and cluster the sentences of a word (Section 4.1). As the next step, we aim to determine the sentence context features that are responsible for the formation of the clusters. For this, we try to recreate the clusters using the sentence context information of the sentences alone. We formulate this task as a classification task. We represent each sentence with a set of sentence context features and we assign the sentences to the clusters based on these features using classifiers. These classifiers are trained to predict the cluster labels from the sentence context features of the sentences (referred to as *sentence-feature classifiers*) (Section 4.2).

This gives us the advantage of representing sentences with discrete features, as opposed to contextualized representations which are continuous.

This enables us to identify the specific sentence context features that contribute to the classifier performance. For this purpose, we use *recursive feature elimination* and we determine the top 10 features that are most important for the performance of the classifiers. Finally, we qualitatively evaluate the significance of the selected features to WSD (Section 4.3).

### 4.1 Clustering

As explained in Section 3, we focus on sense groups and each sense group contains several senses of a word. We perform word sense clustering with each sense group, clustering the sentences of senses within each group.

We cluster the sentences using the word’s CWEs in these sentences. We extract the CWEs from the English BERT model (*base*, *cased*)<sup>3</sup> from each layer.<sup>4</sup> In cases where the words are tokenized into subwords, only the first subword’s embedding is used.

We use the *K-means* clustering algorithm, selecting *k* as the number of senses in each group.<sup>5</sup> We evaluate the performance by comparing cluster labels to the sense labels using *Adjusted Rand Index* (ARI). To be able to compare the performance of each lexical ambiguity type, first, we determine the performance for each sense group within a type, then calculate their average and this average represents the performance of each lexical ambiguity type. We compare the performance change across layers and also the performances based on the best-performing layer. We expect homonymy to perform better in word sense clustering based on the

<sup>3</sup>We choose the BERT *cased* model because it encodes more concepts relevant to WSD, such as words’ WordNet concepts, compared to the *uncased* model, which encodes more linguistic concepts (Sajjad et al., 2022).

<sup>4</sup>We use the Transformers library (Wolf et al., 2020) for extracting the embeddings.

<sup>5</sup>Sci-kit learn library is used for the implementation (Pedregosa et al., 2011).

<sup>2</sup>See Appendix B for the list of words and the number of their senses used in this study.

findings of the previous studies (Nair et al., 2020; Haber and Poesio, 2021).

## 4.2 Sentence-Feature Classifiers

We use the resulting clusters from the previous experiments for training and testing the classifiers. We use the last layer’s results because this layer performs best in the clustering experiment. For each sense group, we train a classifier for cluster assignment: to predict which cluster a sentence belongs to. We only select the sense groups that have more than 25 sentences in each cluster since this experiment requires data for training. This reduces the number of sense groups we focus on in this experiment. Additionally, even though we do not limit the number of senses in each group, we end up with only two senses per group. See Table 1 for the number of sense groups for each experiment.

Our aim is to predict the cluster that each sentence belongs to based on its sentence context features. First, we represent each sentence with a manually selected sentence context features; bag-of-words, morphological properties of the target word (tense, number, etc.), POS tag of the word’s neighbors, the syntactic role of the target word, and the position of the target word in the sentence. We create a sentence feature matrix by binarizing and combining features, resulting in a one-hot representation for each sentence. We select these features to be able to represent the sentences with their context as much as possible. Additionally, we aim to investigate whether the position of the word in the sentence affects the similarity of the representations, considering that positional embeddings are added to the word’s representations with the BERT model.<sup>6</sup>

We process the sentences with the spaCy library<sup>7</sup> to automatically extract this information from the sentences. For the morphological properties of the target word, we use the fine-grained POS tag of the word. Similarly, we use the dependency label of the words as their syntactic role.<sup>8</sup> Bag-of-words representations of the sentences are created by first lemmatizing the sentences, also with spaCy.

For each sense group, we use the sentences in all clusters as our training and test data (split by 3:1). We give each cluster a label (0, 1). For each sen-

tence, the input is its sentence feature matrix and the output is the label of its cluster. We use the linear SVM algorithm to train the classifiers because it is ideal in cases where the number of features is larger than the number of samples. Since each sense group contains two senses in this experiment, our task is to do binary classification to assign the correct cluster label.

We evaluate the performance of the classifiers based on lexical ambiguity type. We calculate the average F1 score (as well as precision and recall scores) for all sense groups within a type and consider it as each type’s performance. The high performance of the classifiers will be an indication that clusters can be recreated with these features and therefore these features can explain the clusters.

Additionally, we train another type of classifiers: classifiers for WSD. These classifiers are trained similarly to the classifiers for cluster assignment; for each sense group and using sentence features as the input. But this time instead of predicting the cluster labels, the classifiers are tasked to predict the sense labels. We compare the performances of the classifiers trained for cluster assignment and WSD. This way, we aim to understand how helpful these features are for WSD to begin with. If the classifiers for cluster assignment perform better than the classifiers for WSD, this can suggest that the clusters are more distinguishable by the sentence context features than the senses and this is already an indication that clusters are formed by the sentence context features that are insignificant to WSD. Additionally, we expect these classifiers to perform better with homonymy compared to polysemy because sentence context is more helpful for the disambiguation of homonymy.

## 4.3 Feature Importance

In order to identify the sentence context features that are responsible for the clusters, we need to identify the features that are important for the performance of the classifiers for cluster assignment. For this purpose, we apply *recursive feature elimination* (RFE) on top of the classifiers.<sup>9</sup> RFE functions as a wrapper feature selection algorithm. It assesses the importance of each feature and iteratively removes the least important ones. The model is then re-fitted with the reduced feature set, and this process continues until the desired number of features is achieved.

<sup>6</sup>For detailed information about the size of the data and the feature matrices for each word, see Appendix D.

<sup>7</sup>Available at: <https://spacy.io/>

<sup>8</sup>See Appendix A for the tags used and their descriptions.

<sup>9</sup>Sci-kit learn library is used both for the implementation of RFE and the training of the classifiers.

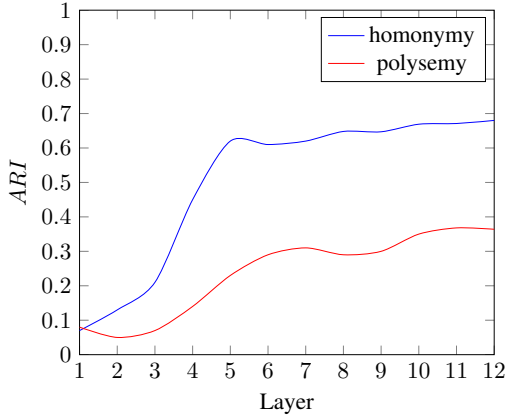


Figure 2: Layer-wise clustering performance with homonymy and polysemy.

We apply RFE to reduce the sentence feature matrix to 10 features. We train the classifiers with RFE using the same training and test datasets as the classifiers trained with the full-sentence feature matrices. By selecting the top 10 features from the sentence feature matrix, we determine which features are important for the correct classification. Then, we evaluate the performance of the classifiers that are trained only with the top 10 features on cluster assignment. Additionally, we train classifiers for cluster assignment with randomly selected 10 features for each sense group in order to establish a baseline. The baseline is determined by averaging the performance of the classifiers across 5 runs for all sense groups. The high performance of the classifiers that are trained with the top 10 features will indicate that these features are responsible for the formation of the clusters.

Finally, we assess the significance of these features to WSD for each word through qualitative evaluation. There are two reasons why we opt for qualitative evaluation. First, there might be coincidental similarities in sentence context within the sentences of one sense that help the clustering process but that are insignificant to WSD. For example, a verb’s most past tense occurrences might coincidentally overlap with one sense, and generalizing over this pattern can help the WSD process. However, relying on these patterns is less than ideal. In such cases, performance-based evaluation cannot effectively capture the significance of these features because they might artificially boost performance. Our primary aim is to uncover these features. The second reason is the limited data size. Qualitative evaluation allows for a deeper understanding, even in situations where the data is limited.

## 5 Results

### 5.1 Clustering

As shown in Figure 2, the clustering performance improves across the layers, with the highest performance observed in the final layer for both types. Word sense clustering performs better with homonymy than polysemy. In the last layer, ARI score is 0.68 for homonymy and 0.36 for polysemy, as shown in Table 1. Regarding the layer-wise performance of the clustering, the pattern for homonymy and polysemy is different. There is a significant performance improvement observed between the 3rd and 5th layers for homonymy. However, there isn’t that steep gain in performance for polysemy overall. This suggests that homonymous senses are mostly disambiguated early in the model layers. Overall, these results are in line with our expectations; the performance with homonymy is higher than with polysemy.

### 5.2 Sentence-Feature Classifiers

The classifiers for WSD achieve an F1 score of 0.80 for homonymy and 0.72 for polysemy. This difference supports our hypothesis; sentence context features are more useful for the disambiguation of homonymy than polysemy. Regarding the classifiers for cluster assignment, there are also performance differences for each lexical ambiguity type. The performance is better with homonymy (0.87) than with polysemy (0.78). Overall, they achieve an F1 score of 0.82.

The classifiers for cluster assignment show better performance compared to the classifiers for WSD, with an overall increase of 0.06 point. There is an increase for both homonymy (0.07) and polysemy (0.05). This increase suggests that the sentence context features are more prominent in the clusters than the original sense sentences and the clusters are more easily distinguishable by their features compared to the senses. Finally, the overall high performance of the classifiers for cluster assignment suggests that the selected sentence context features are a good starting point for feature selection. The results of the classifier performances can be seen in Table 1.

### 5.3 Feature Importance

**Top 10-Feature Classifiers for Cluster Assignment.** The classifiers trained with the top 10 features for cluster assignment achieve good performance with an overall F1 score of 0.85, surpassing

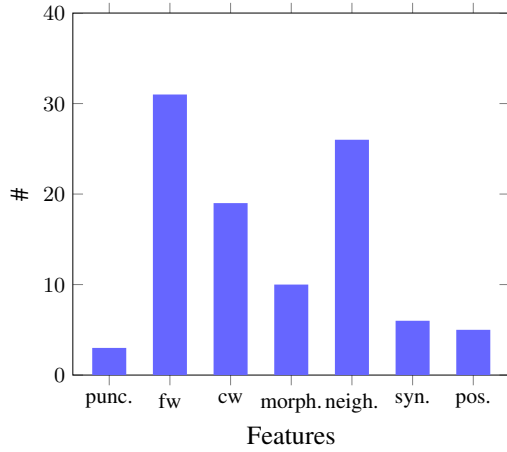


Figure 3: Count of each feature category. The categories are punctuation marks, function words (*fw*), content words (*cw*), neighboring words, the position of the target words in the sentences, and morphological properties and syntactic roles of the target words.

the random baseline (F1: 0.39) by a large margin (see Table 1). This indicates that: *the selected top 10 features are able to recreate the clusters to a great extent.*

**Evaluation of the Selected Features.** We group the top 10 features selected for all sense groups by their similarities and 6 categories are formed. These categories are punctuation marks, function words, content words, POS tags of neighboring words, morphological properties of the target words, syntactic roles of the target words, and positions of the target words in the sentences. Their counts can be seen in Figure 3.<sup>10</sup>

Punctuation marks (‘-’, ‘;’, etc.), function words (‘if’, ‘not’, etc.), and content words (‘river’, ‘bed’, etc.) are the items that are found in the bag-of-words representations of the sentences and are selected as important features. This means that the fact that there are certain items in the sentence determines the decision of the classifiers.

Furthermore, morphological properties of the target word, e.g. whether the verb is in past tense or not, and the syntactic role of the target word, e.g. whether the noun is the direct object of the sentence or not, determine the decision of the classifiers. Similarly, the POS tags of neighboring words also is a determining feature. For example, whether a verb is followed by an adverb or not or whether a verb is followed by a punctuation or not. Finally, the position of the target word is also a determining

feature. However, only the 6th, 7th, 8th, 9th and 10th positions are found to be important.

First, without looking at the details, it is apparent that certain feature categories lack significance or have little significance to WSD. These categories include punctuation marks, the position of the target word in the sentence, the syntactic role and the morphological properties of the target words. On the other hand, features such as POS tags of the neighboring words, and the existence of some content words in the sentence can carry more significance. For example, whether a verb is followed by a preposition or not can be a good indicator of a sense. Similarly, the presence of a word in the sentence can signal one sense, as previously shown in (1) for ‘bank’.

Yet, a closer examination reveals even more striking results. *In most cases, the important features are insignificant to WSD*, except for a few words and this explains the poor clustering performance. The main issue is that most of the time, *a particular insignificant feature is found in both sense sentences and causes these sentences to cluster together.* The features from all categories affect the performance like this. For example, sentences of different senses of a verb are clustered together because, in all of them, the verb is in the past tense, as illustrated in example (3) with two senses of the verb ‘indicate’.

- (3) a. *be a signal for or a symptom of:*  
 “The statistics hardly **indicated** that...”  
 b. *to state or express briefly:*  
 “He **indicated** that requests would...”

Other times, *one feature that is not significant to WSD is found only in the sentences of one sense coincidentally and causes these sentences to cluster together.* While this might affect the performance positively, it does so for reasons that are not ideal. This finding aligns with our expectations. For example, the word ‘other’ is selected as an important feature for the clusters of ‘time’. This feature is not significant to the WSD of this word and it is even not found in direct syntactic relation with the target word in the sentences as in example (4).

- (4) The debris of his **other** careers was piled everywhere; a pile of wire cages for mice from his **time** as a geneticist and a microscope lying on its side on the window sill...

Finally, we do not observe specific patterns for dif-

<sup>10</sup>A detailed list can be seen in Appendix C.

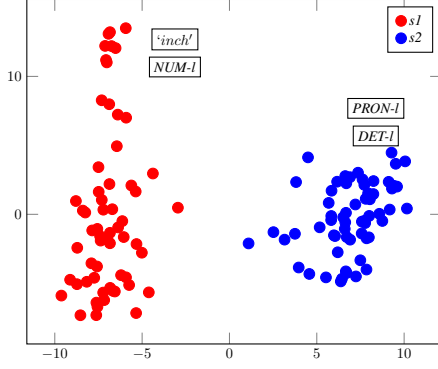


Figure 4: PCA visualization of the embeddings of ‘foot’ in different sense sentences and important features found in each sentence cluster of the word. The embeddings are extracted from the last layer of the BERT model. The features are ‘inch’, *NUM-1* (the left neighbor is a numeral), *PRON-1* (the left neighbor is a pronoun) and *DET-1* (the left neighbor is a determiner).

ferent lexical ambiguity types, however, in general, we observe that for some words, there are more clear clues in the sentence context that are helpful for disambiguation.

**An Example: ‘ask’ vs. ‘foot’.** ‘ask’ in its first sense means “to request something” as in (5-a) and in its second sense means literally “to ask a question” and in this sense, it is also frequently used with direct speech as in (5-b). ‘foot’ in its first sense is the *body part* and in its second sense, it is the *measuring unit*, as illustrated in (6).

- (5) ‘ask’:
- to request something*: “He **asked** her for recommendation.”
  - to ask a question*: “Don’t **ask** a question.”, “‘Who said that?’ he **asked**.”
- (6) ‘foot’:
- body part*: “He hit his **feet**.”
  - measuring unit*: “She is five **feet** tall.”

Even though both of these words are homonymous, there are performance differences between them.<sup>11</sup> Word sense clustering achieves perfect performance with ‘foot’ (1.0) and bad performance with ‘ask’ (0.16). However, the sentence-feature classifiers for cluster assignment perform well with both words (‘ask’: 0.77, ‘foot’: 1). Looking at the classifier performance, we can conclude that the clusters of both words are distinguishable based on their sentence context features. However, looking

<sup>11</sup>See Appendix D for a performance comparison of all words in the last experiment.

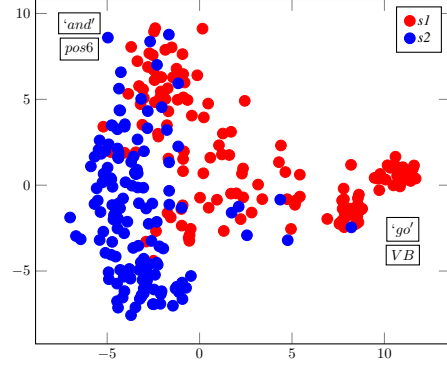


Figure 5: PCA visualization of the embeddings of ‘ask’ in different sense sentences and important features found in each sentence cluster of the word. The embeddings are extracted from the last layer of the BERT model. The features are ‘go’, ‘and’, *VB* (the verb is in base form), and *pos6* (the word is the 6th token in the sentence).

at the word sense clustering performance, we can understand that these features are not equally significant to WSD for both words; they are significant in the case of ‘foot’, but not ‘ask’.

The clusters of ‘foot’ represent each sense well as shown in Figure 4, and features for each cluster are also related to different senses of ‘foot’. For the cluster related to the ‘measuring unit’ sense, the features ‘inch’, and *NUM-1* (the left neighbor is a numeral, as in “5 feet”) are selected as important features. Whereas, for the cluster related to the ‘body part’ sense, the features *PRON-1* (the left neighbor is a pronoun, as in “his feet”), and the feature *DET-1* (the left neighbor is a determiner, as in “the feet”) are selected. These features are indeed good indicators of these senses.

On the other hand, we do not see nicely formed clusters for ‘ask’ (see Figure 5) and we see that the similarity of the representations is driven by the features that are not significant to WSD. For example, all sentences in which the verb is in base form (*VB*) or the word is the 6th token (*pos6*) or the sentences that have the words ‘go’ or ‘and’ are clustered together.

Two senses of ‘ask’ occur in different sentence structures: the first sense occurs with prepositional objects, as in (5-a), and the second sense with direct speech as in (5-b). It is interesting to see that these distinctions are not captured by the clusters and these features do not determine the similarity of the representations. We also see that the sentence-feature classifier for WSD performs better (0.81) than the sentence-feature classifier for cluster as-

signment (0.77) with ‘ask’. This contrasts with the general pattern. This might indicate that the senses are actually distinguishable by their sentence context features, however, not these features but insignificant features are responsible for the formation of the clusters.

## 6 Discussion

In order to identify the sentence context features that are responsible for the similarity of the contextualized representations, we conducted a cluster explainability study and identified the sentence context features that lead to the formation of the clusters in word sense clustering with CWEs. Our results have shown that features from different categories determine the similarity of the representations; function words, punctuation marks, content words in the sentences, position of the target word in the sentence, neighboring words, morphological properties and the syntactic role of the target word. Our results are in line with [Sajjad et al. \(2022\)](#) who have shown that the CWEs encode both grammatical and semantic properties of the words and the clusters of CWEs reveal these similarities.

Furthermore, we qualitatively evaluated the identified features for each word and have shown that they are mostly insignificant to WSD. We observed that even when different senses of a word occur in diverse contexts and the sentence context provides clear clues for different senses (as in the case with ‘ask’), the significant features do not determine the similarity of the representations in most cases. This contradicts our expectations. When the sentence context provides clear clues for different senses, e.g. in the case of homonymy, we expected the similarity of contextualized representations to be determined by these clues. However, this is not the case and there are other features in the sentences that are insignificant to WSD, that affect the similarity of the representations more.

Our analysis also has revealed that insignificant features affect the clustering performance negatively in several ways. Most commonly, some insignificant features occur in the sentences of both word senses and they lead these sentences to cluster together. This explains the poor clustering performance reported previously ([Yenicelek et al., 2020](#)) and also in this study. Additionally, in some cases, certain insignificant features occur only in the sentences of one sense by chance and they lead to the formation of clusters. Although these cases don’t

affect the performance negatively, this shows how the randomness in the data can affect the clustering performance.

In relation to the performance with different lexical ambiguity types, the findings of our study are in line with previous studies ([Nair et al., 2020](#); [Haber and Poesio, 2021](#); [Sevastjanova et al., 2021](#)). Clustering performs better with homonymy than polysemy. In addition to previous studies, our results have shown that homonyms are more distinguishable by sentence context features than polysemes and their disambiguation can be more easily achieved with a simple classifier trained with these features. However, contextualized representations’ similarity is not consistently determined by the sense-significant features even for homonyms.

## 7 Conclusion

The information encoded in contextualized representations which determines their similarity is not significant to WSD in most cases. This shows that these representations do not capture the different meanings of words as expected, explaining why using CWEs of pre-trained language models alone does not yield sufficient performance in unsupervised WSD. In the future, we plan to explore possible strategies to create contextualized representations that are more suitable to WSD by limiting the information that is insignificant to WSD encoded in the representations. This way, we aim to enhance unsupervised WSD performance.

## 8 Acknowledgement

This study is funded by the project “Coercion and Copredication as Flexible Frame Composition” funded by DFG (Deutsche Forschungsgemeinschaft). We would like to thank the anonymous reviewer for their valuable comments. Lastly, special thanks to David Arps for interesting and stimulating discussions.

## References

- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [BERT: Pre-training of deep bidirectional transformers for language understanding](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

- Kawin Ethayarajh. 2019. [How contextual are contextualized word representations? Comparing the geometry of BERT, ELMo, and GPT-2 embeddings](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 55–65, Hong Kong, China. Association for Computational Linguistics.
- Christiane Fellbaum. 2010. Wordnet. In *Theory and applications of ontology: computer applications*, pages 231–243. Springer.
- Marcos Garcia. 2021. [Exploring the representation of word meanings in context: A case study on homonymy and synonymy](#). In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3625–3640, Online. Association for Computational Linguistics.
- Luke Gessler and Nathan Schneider. 2021. [BERT has uncommon sense: Similarity ranking for word sense BERTology](#). In *Proceedings of the Fourth BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 539–547, Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Janosch Haber and Massimo Poesio. 2021. [Patterns of polysemy and homonymy in contextualised language models](#). In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2663–2676, Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Ekaterini Klepousniotou. 2002. [The processing of lexical ambiguity: Homonymy and polysemy in the mental lexicon](#). *Brain and Language*, 81(1):205–223.
- Ekaterini Klepousniotou and Shari R. Baum. 2007. [Disambiguating the ambiguity advantage effect in word recognition: An advantage for polysemous but not homonymous words](#). *Journal of Neurolinguistics*, 20(1):1–24.
- Ekaterini Klepousniotou, G. Bruce Pike, Karsten Steinhauer, and Vincent Gracco. 2012. [Not all ambiguous words are created equal: An eeg investigation of homonymy and polysemy](#). *Brain and Language*, 123(1):11–21.
- Ekaterini Klepousniotou, Debra Titone, and Carolina Romero. 2008. Making sense of word senses: the comprehension of polysemy depends on sense overlap. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 34(6):1534.
- Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. [Building a large annotated corpus of English: The Penn Treebank](#). *Computational Linguistics*, 19(2):313–330.
- Timothee Mickus, Denis Paperno, Mathieu Constant, and Kees van Deemter. 2020. [What do you mean, BERT?](#) In *Proceedings of the Society for Computation in Linguistics 2020*, pages 279–290, New York, New York. Association for Computational Linguistics.
- George A. Miller, Claudia Leacock, Randee Tengi, and Ross T. Bunker. 1993. [A semantic concordance](#). In *Human Language Technology: Proceedings of a Workshop Held at Plainsboro, New Jersey, March 21-24, 1993*.
- Sathvik Nair, Mahesh Srinivasan, and Stephan Meylan. 2020. [Contextualized word embeddings encode aspects of human-like word sense knowledge](#). In *Proceedings of the Workshop on the Cognitive Aspects of the Lexicon*, pages 129–141, Online. Association for Computational Linguistics.
- Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajič, Christopher D. Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, Reut Tsarfaty, and Daniel Zeman. 2016. [Universal Dependencies v1: A multilingual treebank collection](#). In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC’16)*, pages 1659–1666, Portorož, Slovenia. European Language Resources Association (ELRA).
- Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. 2011. [Scikit-learn: Machine learning in python](#). *Journal of Machine Learning Research*, 12(85):2825–2830.
- Hassan Sajjad, Nadir Durrani, Fahim Dalvi, Firoj Alam, Abdul Khan, and Jia Xu. 2022. [Analyzing encoded concepts in transformer language models](#). In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3082–3101, Seattle, United States. Association for Computational Linguistics.
- Rita Sevastjanova, Aikaterini-Lida Kalouli, Christin Beck, Hanna Schäfer, and Mennatallah El-Assady. 2021. [Explaining contextualization in language models using visual analytics](#). In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 464–476, Online. Association for Computational Linguistics.
- Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame,

Quentin Lhoest, and Alexander Rush. 2020. [Transformers: State-of-the-art natural language processing](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

David Yenicecik, Florian Schmidt, and Yannic Kilcher. 2020. [How does BERT capture semantics? a closer look at polysemous words](#). In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 156–162, Online. Association for Computational Linguistics.

| Tag      | Description                       |
|----------|-----------------------------------|
| nsubj    | nominal subject                   |
| pobj     | object of a preposition           |
| attr     | attribution                       |
| xcomp    | open clausal complement           |
| npadvmod | noun phrase as adverbial modifier |

Table 2: Dependency labels used in this study, from spaCy model *en\_core\_web\_trf*.

| Tag | Description                           |
|-----|---------------------------------------|
| NN  | Noun, singular or mass                |
| NNS | Noun, plural                          |
| VBD | Verb, past tense                      |
| VBP | Verb, non-3rd person singular present |
| VBZ | Verb, 3rd person singular present     |
| VB  | Verb, base form                       |

Table 3: Fine-grained POS tags used in this study, from Penn Tree Bank (Marcus et al., 1993).

| Tag   | Description               |
|-------|---------------------------|
| ADP   | adposition                |
| PUNCT | punctuation               |
| PART  | particle                  |
| SCONJ | subordinating conjunction |
| PRON  | pronoun                   |
| DET   | determiner                |
| ADV   | adverb                    |
| NUM   | numeral                   |

Table 4: POS tags used in this study, from Universal Dependencies (Nivre et al., 2016).

## A Feature Tags

The feature tags for the morphological properties of the words, the syntactic role of the words, and the POS tags of neighboring words that are used

|                    |                |                 |
|--------------------|----------------|-----------------|
| <b>ask: 1</b>      | heart: 2       | <b>life: 5</b>  |
| <b>begin: 3</b>    | produce: 2     | point: 5        |
| degree: 3          | put: 3         | raise: 3        |
| drive: 3           | table: 2       | <b>time: 2</b>  |
| <b>foot: 2</b>     | right: 2       | <b>way: 4</b>   |
| heart: 2           | case: 3        | <b>world: 4</b> |
| <b>indicate: 2</b> | consider: 4    | plane: 2        |
| light: 3           | cover: 2       | lead: 7         |
| <b>man: 3</b>      | <b>door: 2</b> |                 |

Table 5: The words that are found in our dataset with their sense counts. All the words are used in the word sense clustering and the bold words are used in the cluster explainability experiment.

in this study can be found in Table 2, 3, 4. For the morphological properties of the words, we use their fine-grained POS tag (Table 3). For the syntactic role of the words, we use their dependency label (Table 2). All the labels are obtained by processing the sentences with spaCy.

## B Selected Words

A list of the words that are found in our dataset can be seen in Table 5. All the words and senses are used in the word sense clustering experiment. Only 10 words and 2 sense each are used in the cluster explainability experiment.

## C Selected Features List

A detailed list of selected features from each category can be seen in Table 6.

## D Performance with Individual Words

The individual performance of each word can be seen in Table 7. Only the performances of the words that are used in the last experiment are reported. Both clustering performance and the performance of the sentence-feature classifiers are reported. The data size (number of sentences) of each word can be also seen in Table 7.

| Feature Category          | Features                                                                                                                                                                                                                                                                                  |
|---------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Neighbouring Word (right) | ADP: 3, PUNCT: 2, PART: 2, CONJ: 1, PRON: 1, DET: 1, ADV: 1                                                                                                                                                                                                                               |
| Neighbouring Word (left)  | DET: 4, PART: 2, PRON: 2, NUM: 2, ADP: 2, ADJ: 2, PUNCT: 1                                                                                                                                                                                                                                |
| Punctuation               | '.' : 2, '-' : 1,                                                                                                                                                                                                                                                                         |
| Function Word             | 'and': 1, 'after': 1, 'can': 1, 'she': 1, 'might': 1, 'the': 3, 'to': 2, 'when': 1, 'my': 1, 'on': 1, 'through': 1, 'no': 1, 'with': 1, 'just': 1, 'a': 1, 'by': 1, 'however': 1, 'in': 1, 'or': 1, 's': 1, 'each': 2, 'which': 1, 'her': 1, 'other': 1, 'their': 1, 'once': 1, 'such': 1 |
| Content Word              | 'mean': 1, 'nothing': 1, 'high': 1, 'outside': 1, 'see': 1, 'first': 1, 'route': 1, 'Spencer': 1, 'new': 1, 'plan': 1, 'event': 1, 'God': 1, 'three': 1, 'hand': 1, 'go': 1, 'take': 1, 'inch': 1, 'age': 1, 'feel': 1                                                                    |
| Syntactic Role            | nsubj: 2, pobj: 1, xcomp: 1, attr: 1, npadvmod: 1                                                                                                                                                                                                                                         |
| Morphological Properties  | NNS: 3, VBD: 2, NN: 2, VBP: 1, VBZ: 1, VB: 1                                                                                                                                                                                                                                              |
| Word Position             | 6th: 1, 7th: 1, 8th: 1, 9th: 1, 10th: 1                                                                                                                                                                                                                                                   |

Table 6: Selected important features in each category and their counts. POS tags of the word’s neighbors are given for neighboring words, the dependency label of the word is given for the syntactic role, and the fine-grained POS tag is given for the word’s morphological properties. See Appendix A for the descriptions of the tags used.

| Word            | Data# | Clustering Performance | Feature# | Sentence-Feature Classifiers |          |          |             |          |          |             |          |          |
|-----------------|-------|------------------------|----------|------------------------------|----------|----------|-------------|----------|----------|-------------|----------|----------|
|                 |       |                        |          | WSD                          |          |          | Cluster     |          |          | 10-Cluster  |          |          |
|                 |       |                        |          | F1                           | P        | R        | F1          | P        | R        | F1          | P        | R        |
| <i>Homonymy</i> |       |                        |          |                              |          |          |             |          |          |             |          |          |
| ask             | 278   | 0.16                   | 1668     | <b>0.81</b>                  | 0.82     | 0.81     | 0.77        | 0.77     | 0.77     | 0.77        | 0.77     | 0.77     |
| begin           | 90    | 0.34                   | 1007     | 0.87                         | 0.87     | 0.87     | <b>0.91</b> | 0.93     | 0.92     | 0.80        | 0.80     | 0.80     |
| foot            | 119   | 1                      | 1008     | <b>1</b>                     | <b>1</b> | <b>1</b> | <b>1</b>    | <b>1</b> | <b>1</b> | <b>1</b>    | <b>1</b> | <b>1</b> |
| indicate        | 70    | 0.12                   | 809      | 0.62                         | 0.62     | 0.62     | <b>0.85</b> | 0.86     | 0.85     | <b>0.85</b> | 0.89     | 0.85     |
| man             | 92    | 0.54                   | 932      | 0.71                         | 0.71     | 0.70     | 0.82        | 0.83     | 0.82     | <b>0.85</b> | 0.86     | 0.85     |
| <i>Polysemy</i> |       |                        |          |                              |          |          |             |          |          |             |          |          |
| life            | 74    | 0.34                   | 775      | 0.59                         | 0.60     | 0.60     | 0.81        | 0.83     | 0.81     | <b>0.86</b> | 0.86     | 0.86     |
| man             | 54    | 0.16                   | 690      | 0.66                         | 0.81     | 0.68     | 0.84        | 0.84     | 0.84     | <b>0.94</b> | 0.95     | 0.94     |
| time            | 56    | 0.69                   | 762      | 0.71                         | 0.76     | 0.71     | 0.84        | 0.84     | 0.84     | <b>0.94</b> | 0.95     | 0.94     |
| way             | 110   | 0.55                   | 1179     | 0.81                         | 0.81     | 0.81     | <b>0.86</b> | 0.86     | 0.86     | 0.73        | 0.74     | 0.73     |
| world           | 56    | 0.35                   | 692      | <b>0.84</b>                  | 0.84     | 0.84     | 0.53        | 0.66     | 0.55     | 0.83        | 0.88     | 0.83     |

Table 7: Data size and performance details of individual words. *Data#* refers to the number of sentences. Clustering performance is evaluated using ARI. *Feature#* is the size of the sentence feature matrix. *WSD* refers to the classifiers trained for WSD, *Cluster* for cluster assignment. *10-Cluster* classifiers refer to the classifiers trained for cluster assignment with only the top 10 features. *F1*, *Precision* and *Recall* scores are given for each classifier. The best F1 score for each word is given in bold.



## Chapter 5

# Improving Word Sense Induction through Adversarial Forgetting of Morphosyntactic Information

# Improving Word Sense Induction through Adversarial Forgetting of Morphosyntactic Information

Deniz Ekin Yavas<sup>\*1</sup>, Timothée Bernard<sup>2</sup>, Laura Kallmeyer<sup>1</sup>, Benoît Crabbé<sup>2</sup>

Heinrich Heine University Düsseldorf<sup>1</sup>, Université Paris Cité<sup>2</sup>

{deniz.yavas, laura.kallmeyer}@hhu.de<sup>1</sup>

{timothee.bernard, benoit.crabbe}@u-paris.fr<sup>2</sup>

## Abstract

This paper addresses the problem of word sense induction (WSI) via clustering of word embeddings. It starts from the hypothesis that contextualized word representations obtained from pre-trained language models (LMs), while being a valuable source for WSI, encode more information than what is necessary for the identification of word senses and some of this information affect the performance negatively in unsupervised settings. We investigate whether using contextualized representations that are invariant to these ‘nuisance features’ can increase WSI performance. For this purpose, we propose an adaptation of the adversarial training framework proposed by Jaiswal et al. (2020) to erase specific information from the representations of LMs, thereby creating *feature-invariant representations*. We experiment with erasing (i) morphological and (ii) syntactic features. The results of subsequent clustering for WSI show that these features indeed act like noise: Using feature-invariant representations, compared to using the original representations, increases clustering-based WSI performance. Furthermore, we provide an in-depth analysis of how the information about the syntactic and morphological features of words relate to and affect WSI performance.

## 1 Introduction

Words in their different senses occur in different contexts. Contextualized word representations obtained from transformer based pre-trained language models (LMs) such as BERT (Devlin et al., 2019) are especially suitable for Word Sense Disambiguation (WSD) because they capture the sentential context of a word and thereby oftentimes allow to distinguish different senses of a word. They have indeed been successfully used for WSD in recent work (Hadiwinoto et al., 2019; Loureiro et al., 2021; Vandenbussche et al., 2021).

However, in both unsupervised WSD, where the goal is to identify the instances of a specific sense, and word sense induction (WSI), which allows the discovery of novel senses, using the LM representations alone does not yield satisfactory results (Pilehvar and Camacho-Collados, 2019; Samih and Kallmeyer, 2023). In both, similarity of the representations plays a crucial role, and this similarity is determined by many features. Indeed, the contextualized representation of a word usually encodes a wide range of linguistic information about the word in its context, such as its syntactic function, its morphological properties, its position, its casing, and the identity of its neighbouring words (Sajjad et al., 2022). However, most of the encoded information that determines the similarity of the representations is not relevant to word senses (Yavas, 2024). Note that this is not a problem for supervised WSD, as a supervised model can learn to ignore those features that are not discriminative for word senses.

Building on these insights, we focus on WSI and aim at investigating the relationship between specific types of information encoded in contextualized representations of LMs and WSI performance. Concretely, we examine whether erasing certain information from the representations of LMs can lead to an increase in performance for a simple clustering-based WSI system. Our investigation examines two types of information that have been observed to affect the word sense clustering performance negatively. Yavas (2024) have shown that in word sense clustering with BERT representations on SemCor (Miller et al., 1993), word instances are frequently clustered together based on the similarities between their morphological and syntactic features (more specifically, *syntactic role* of the word) rather than their semantic similarities. For example, past tense instances of a specific verb, or all instances of a specific noun occurring as direct objects, are clustered together.

We adapt the adversarial training framework of

<sup>\*</sup>This work was conducted during the author’s visit to Université Paris Cité.

Jaiswal et al. (2020) in order to train a forget-gate that erases information from the representations of LMs, resulting in *feature-invariant representations*. We experiment with the BERT model and create feature-invariant representations for both of the above-mentioned types of features (morphological and syntactic). Finally, we evaluate the performance of WSI on SemCor with different feature-invariant representations, comparing them to the original word representations obtained from BERT. Furthermore, we conduct an in-depth analysis of how the information about the syntactic and morphological features of words relate to and affect WSI performance.<sup>1</sup>

Our results show that words’ morphological and syntactic features indeed act like noise that negatively affects clustering performance and syntax- and morphology-invariant representations are better suited to WSI than the original BERT representations. Furthermore, a more detailed analysis of the relation between these information types and WSI performance shows that even though syntactic features are more correlated to word senses than morphological features are, they still affect the WSI performance negatively overall.

This paper makes several contributions. First, we propose an adaptation of the framework proposed by Jaiswal et al. (2020) to erase unwanted information from the representations of LMs. Secondly, we use this method to generate syntax- and morphology-invariant representations from the word representations of the BERT model and achieve better performance in clustering-based WSI. Lastly, we provide an in-depth analysis of how the morphological and syntactic features of words affect WSI performance.

The paper is structured as follows: We review related work in Section 2, then we introduce our framework for creating feature-invariant representations in Section 3 and report the results of the creation process. Finally in Section 4, we report the experiments on WSI with an analysis of the relation between the information types and WSI performance.

<sup>1</sup>We also experimented with positional information. In our experiments, we successfully removed the positional information from the representations, however, these representations exhibited unexpected behaviour in clustering experiments. As a result, we decided to exclude this feature type. We intend to investigate the underlying reasons in the future.

## 2 Related Work

**Word Sense and Information Encoded in Contextualized Representations.** Contextualized representations of pre-trained LMs encode more contextual information than what is necessary for the identification of word senses and this information can affect the similarity of the representations in an unwanted way. Sajjad et al. (2022) have shown that semantic, morphological, and syntactic concepts are encoded in contextualized representations. These concepts include words’ POS tags, CCG super-tags, ngrams, casings, WordNet concepts, and so on. Furthermore, clustering of contextualized word representations reveal these similarities between the words.

In their detailed qualitative analysis, Yavas (2024) have shown that word sense clustering with BERT’s representations on SemCor is heavily and negatively affected by information encoded in the representations from the sentence context that is insignificant to WSD, such as some morphological features of the words, their syntactic role, the presence of some punctuation marks and function words in the sentence (e.g. ‘not’, ‘then’, etc.). In the present study, we aim to investigate whether the effects of some of these features can be controlled and whether doing so can increase performance in WSI on the same dataset.

Similar effects have been found in lexical semantic change detection. Laicher et al. (2021) have observed that BERT representations are influenced by the orthographic form of words. Consequently, this affects how the representations are clustered. They have shown that removal of this bias increases the clustering performance. In order to do so, they preprocess the input data by lemmatizing the target word in each sentence before feeding it to the model.

**Adversarial Training for Invariant Representation Learning.** Invariant representation learning aims to create representations that do not encode certain unwanted features of data, such as nuisances, biases, or domain-specific features. Nuisances are features in the data that have no or little relevance to the task but influence model performance, like facial expressions in face recognition (Bronstein et al., 2003) or orientation in image recognition (Khotanzad and Hong, 1990). The creation of representations invariant to nuisances aims to increase model performance and robustness.

In this study, we consider morphological and

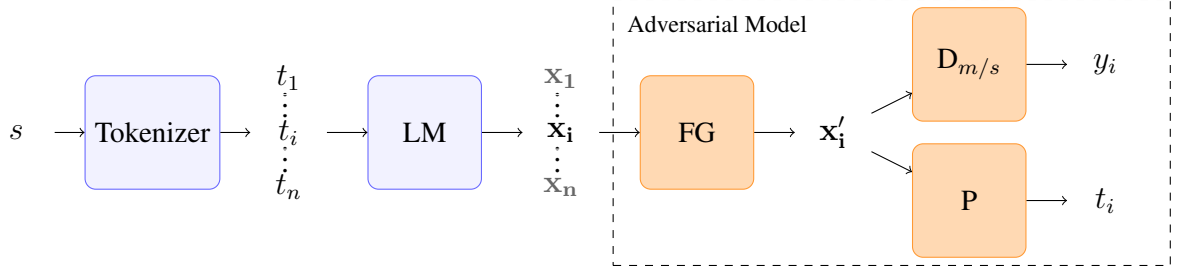


Figure 1: Framework for training a forget-gate ( $FG$ ) to create feature-invariant representations from the representations of a pre-trained model. The  $FG$  is trained as a part of an adversarial model. The  $LM$  is only used as a feature extractor and its parameters are frozen.

syntactic features of words as nuisance features for WSI because they are not directly related to different senses of words but affect performance. We acknowledge that the syntactic features of words are, to some extent, relevant to WSI, but while these features may aid in identifying the senses of some words, we hypothesize that it introduces noise overall.

Invariant representation learning is widely used to create representations that are invariant to nuisance features in Computer Vision (Louizos et al., 2017; Xie et al., 2017; Jaiswal et al., 2018, 2020). However, in NLP, most applications of this technique center around learning domain-invariant representations (Louizos et al., 2017; Jaiswal et al., 2018; Peng and Zhang, 2020; Xin et al., 2022). As for our knowledge, there has been no attempt to create contextualized representations invariant to any linguistic information.

Jaiswal et al. (2020) propose a framework for learning invariant representations through adversarial training in a Computer Vision context. They train an encoder (and a decoder) to generate representations for a set of inputs. At the same time, a *forget-gate* is trained to generate masks meant to be applied to the representations in order to create invariant representations. The forget-gate is trained as part of an adversarial model, in which a *discriminator* predicts the unwanted information from the masked representation while a *predictor* predicts some task-related information. Our framework for learning invariant representations is inspired by Jaiswal et al. (2020) while showing clear differences. We do not train an encoder-decoder model but, instead, we utilize LMs and create invariant representations from their representations. Furthermore, forgetting is not done by masking but by transforming the LM representations through a feed-forward network.

### 3 Creating Feature-Invariant Representations via Adversarial Training

In order to obtain contextualized representations that are invariant to certain features, we propose to add a forget-gate on top of a pre-trained LM. The forget-gate applies a nonlinear transformation and thereby selectively removes the unwanted information from the original contextualized representations. It is trained as a part of an adversarial model inspired by Jaiswal et al. (2020). Concretely, we train two forget-gates to create feature-invariant representations for syntactic and morphological features. We will refer to the respective resulting representations as *syntax-invariant*, and *morphology-invariant representations*.

#### 3.1 Framework

Our framework is illustrated in Figure 1. We define a neural network named *forget-gate* ( $FG$ ). This network is implemented as a feedforward neural network with two hidden layers with ReLU activation function. It applies a transformation to the representations obtained from a pre-trained LM to create representations that are invariant to specific information. The input of  $FG$  ( $\mathbf{x}$ ) is the representation we aim to transform, namely the token embedding from the pre-trained LM. The output of  $FG$  ( $\mathbf{x}'$ ) is the feature-invariant representation.

Given a sentence  $s$ , we first tokenize it with the LM tokenizer and then pass the tokenized sentence ( $t_1, \dots, t_i, \dots, t_n$ ) to the LM in order to extract the token embeddings ( $\mathbf{x}_1, \dots, \mathbf{x}_i, \dots, \mathbf{x}_n$ ), obtained from the last layer of the LM (i.e., the LM acts as a feature extractor). Each of these embeddings constitutes one input data point to the  $FG$ . We use the BERT (*base-cased*) model.<sup>2</sup>

<sup>2</sup>The embeddings are extracted using the Transformers library (Wolf et al., 2020).

The forget-gate  $FG$  is trained as part of an adversarial model with two auxiliary modules; a discriminator ( $D$ ) and a predictor ( $P$ ). During training, the representation produced by the forget-gate ( $\mathbf{x}'_i = FG(\mathbf{x}_i)$  for token  $t_i$ ) is given to  $P$  and  $D$ .  $D$  is tasked with probing for unwanted information (some label  $y_i$  for token  $t_i$ ) in the embedding  $\mathbf{x}'_i$ , and  $P$  is tasked with recovering the identity of the token ( $t_i$ ) from  $\mathbf{x}'_i$ . The adversarial model is trained on the representations of both masked and unmasked tokens (i.e., we sometimes substitute the [MASK] token for  $t_i$  in the input of the LM).

The training of the adversarial model alternates between three types of batch, each batch containing training data for only one of the three subnetworks of the adversarial model. On the first type of batch, the parameters of  $D$ , on the second type of batch, the parameters of  $P$ , and on the third type of batch, the parameters of  $FG$  are updated. There are two batches of the first type (for  $D$ ) for one batch of the second and one batch of the third type. So,  $D$  is trained more than the rest of the network. The parameters of  $FG$  are updated based on the combined loss  $L_{FG}$  of  $D$  and  $P$  as indicated in (1). The loss of  $D$  is given as negative since we want to increase it.<sup>3</sup>

$$L_{FG}(x, y, t) = -L_D(D(x'), y) + L_P(P(x'), t) \quad (1)$$

For each feature type (morphological and syntactic), we train an adversarial model with a unique discriminator to obtain a feature-specific forget-gate.<sup>4</sup> The discriminators,  $D_m$  and  $D_s$ , are trained as classification models and towards labelling tokens with POS tags (from the Penn Treebank tagset, Marcus et al., 1993 — these tags are fine-grained and provide morphological information such as number for nouns and tense for verbs) and (incoming) dependency labels respectively; the training labels are predicted, see Section 3.3. For each feature type, the corresponding discriminator aims to probe for this specific feature, while the forget-gate simultaneously aims to erase it. The details about the architecture of the different modules of

the adversarial model and their loss functions can be found in Appendix A.

We train the adversarial models for 800 epochs creating checkpoints every 100th epoch and select the best checkpoint a posteriori based on the evaluation results (see next section). The details about the hyperparameters and the training of the adversarial model can be found in Appendix B. As a result of training the two adversarial models, we obtain two different forget gates,  $FG_m$  and  $FG_s$ . These forget gates, when applied to a BERT word embedding, yield the respective feature-invariant representations.

### 3.2 Evaluation

In order to evaluate this method, we create representations using the trained forget-gates for each type of information and use these representations to train two models from scratch: one for word (i.e., token) prediction and the other for unwanted information probing. The performance of these models on the test data tells us whether the feature-invariant representation creation was successful.

We compare the performances of these models to a lower and an upper bounds. The upper bound for a task is defined as the performance of a similar system trained using the original BERT representations. The lower bound is defined differently for the two types of information. For syntactic information, the probing model is trained using the non-contextualized word representations used by BERT as input to its first layer. For morphological information, the lower bound is given by the most frequent POS baseline. It is calculated for each grammatical category (i.e. noun, verb, and so on.) by predicting the most frequent POS tag for that category and averaged for all categories.

The probing models for morphological and syntactic features are similar to the respective discriminators in the adversarial models: They share the same architecture, training and test data. The word predictors are also similar to the respective predictors in the adversarial models, in regard to their architecture, training and test data.

We compare the performances of different models on the test data. We use perplexity as the metric to evaluate the word predictors and accuracy for the probing models. We select the best forget-gate for each feature type aiming at a low probing performance (close to the lower bound): We evaluate all checkpoints and pick the forget-gate with the lowest probing scores (if not lower than the lower

<sup>3</sup>In Jaiswal et al. (2020)’s framework, forgetting is not done by using the opposite of  $D$ ’s loss on the correct labels, but by using  $D$ ’s loss on random labels.

<sup>4</sup>We target morphological and syntactic information independently, even though theoretically, they are interrelated. However, this does not affect the relevance of our method, but only some linguistic interpretations of the results.

bound). Details about the hyperparameters and the training of the lower bound and upper models, evaluation models (probing and word prediction), and the selected forget-gates are given in Appendix B.<sup>5</sup>

### 3.3 Data

For the training and evaluation of the models, we use the Brown corpus (Kučera et al., 1967). We extract the token representations by BERT of each sentence. These token representations are then used as the input for the forget-gate. Since words are tokenized into WordPiece subwords by the BERT tokenizer, we work with these subwords rather than entire words.

In cases a word is split in multiple parts, we only take the first subword into account, i.e. we only erase information from the first subword a word and we only use this subword for WSI. We expect the first subwords to encode more relevant information for WSI because they are more likely to align with the stems of the words as opposed to suffixes (e.g. ‘booklets’ is tokenized into ‘booklet’ and ‘s’ by the model tokenizer).

We assign two labels to each token; one for the discriminator (or probe) and another for the predictor. The predictor’s label corresponds to the token ID assigned by the BERT’s tokenizer to the token. The discriminator’s label varies depending on the feature type: the label is either the label of the incoming syntactic dependency or the POS tag (of the word associated with the token). We get these labels automatically using spaCy.<sup>6</sup>

The dataset for morphological information only contains tokens of words belonging to grammatical categories that exhibit inflection in English: nouns, verbs, adjectives, adverbs, and pronouns. No such restrictions apply to syntactic information. This process yields datasets containing 2,341,954 tokens for syntactic information, and 1,315,988 tokens for morphological information. All datasets are split to train, development, and test data with the ratio 80:10:10.

### 3.4 Results

Both feature-invariant representations achieve good results in word prediction and probing tasks; the unwanted information is erased from the representations while their word prediction capabilities are

<sup>5</sup>The code for this project is available at: <https://github.com/yavasde/Adversarial-Forgetting-of-Morphosyntactic-Information>.

<sup>6</sup><https://spacy.io/>, model: *en\_core\_web\_trf*.

|                                  | Word Prediction | Probing |
|----------------------------------|-----------------|---------|
| <i>Syntactic Information</i>     |                 |         |
| Upper Bound                      | 3.0             | 85.0    |
| Lower Bound                      | -               | 70.2    |
| Invariant Rep.                   | 4.1             | 72.1    |
| <i>Morphological Information</i> |                 |         |
| Upper Bound                      | 3.0             | 89.1    |
| Lower Bound                      | -               | 62.1    |
| Invariant Rep.                   | 7.1             | 75.9    |

Table 1: Evaluation results for the feature-invariant representations with comparison to the upper and lower bounds of the tasks. *Accuracy* is given for probing (in this context, lower is better) and *perplexity* is given for word prediction results (lower is better).

intact. The lower bounds and upper bounds for all tasks and the evaluation results for feature-invariant representations can be seen in Table 1.

Erasing morphological information impacts the performance of word prediction more. This is expected because the morphological features of words (for instance grammatical number for nouns) are strongly correlated with their word forms.

## 4 Word Sense Induction Performance

Our aim is to investigate whether using feature-invariant contextualized representations can improve WSI performance. For this purpose, we compare the performance on WSI of three variants of the same system, respectively using three different representations; 1) the original contextualized representations of the BERT model, 2) syntax-invariant, and 3) morphology-invariant contextualized representations, where the latter two are obtained by applying our trained forget-gates  $FG_m$  and  $FG_s$  to the BERT representations. Furthermore, we provide a detailed analysis of the relation between the morphological and syntactic features of words and WSI and how the erasure of this information affects the WSI performance.

### 4.1 Data

We evaluate our WSI systems on SemCor. SemCor is based on a subset of the Brown Corpus and it provides sentences in which a word, the *target word*, is labelled with a WordNet sense (Fellbaum, 1998) as shown in (1). We focus on nouns and verbs and exclude other grammatical categories. We further exclude the words that have only one sense, and the senses that occur in less than 10 sentences.

- (1) officer:
- a. “An officer with a squad of men had been waiting on the bank.”  
(*officer.n.01*)
  - b. “And the policy officer has the hounds of time snapping at his heels.”  
(*officer.n.02*)

One of the advantages of using SemCor for WSI is that it is a subset of a bigger corpus (Brown Corpus), that we can use to train the forget-gates. The forget-gates are then trained on the same kind of texts that the ones used for WSI, which helps ensuring the quality of the invariant representations used during clustering. There is no methodological problem in doing so as the gold clusters are not used at any time during the training of the forget-gates. This approach can be applied to any dataset by training a forget-gate and performing WSI on the same data. Note that while the training of the forget-gates requires feature annotation, this does not limit the applicability of our approach as we perform it automatically.

## 4.2 Method

We cluster instances of words using their representations (BERT, syntax-invariant or morphology-invariant) in the sentences. We tokenize each sentence with the BERT tokenizer and give the tokenized sentence to the model to extract the representations of the target word from the last layer of the BERT model. In cases where the words are tokenized into subwords, we only use the first subword token. We create feature-invariant representations from the original representations of BERT for each information type using the information-specific forget-gate ( $FG_s$  or  $FG_m$ ). We apply normalization to all embeddings before clustering.

We use the K-Means algorithm for clustering.<sup>7</sup> K-Means requires the cluster number as a hyperparameter. To determine the optimal number of clusters for each word, we run the algorithm with different cluster numbers between 2 and 6 and select the one with the highest silhouette score.<sup>8</sup>

We evaluate the clustering performance by comparing cluster assignments and the WordNet senses of word instances and average the result over all

<sup>7</sup>The algorithms are implemented using the Scikit-learn library (Pedregosa et al., 2011).

<sup>8</sup>The silhouette score measures how similar a sample is to its cluster compared to other clusters. It’s calculated for each sample and then averaged for the entire dataset.

|           | Overall                             | nouns                               | verbs                               |
|-----------|-------------------------------------|-------------------------------------|-------------------------------------|
| BERT      | 0.210 ( $8 \times 10^{-4}$ )        | 0.251 ( $1 \times 10^{-3}$ )        | 0.174 ( $1 \times 10^{-3}$ )        |
| Syn-Inv   | <b>0.221</b> ( $1 \times 10^{-3}$ ) | <b>0.263</b> ( $4 \times 10^{-4}$ ) | <b>0.185</b> ( $2 \times 10^{-3}$ ) |
| Morph-Inv | <b>0.232</b> ( $1 \times 10^{-3}$ ) | <b>0.267</b> ( $1 \times 10^{-3}$ ) | <b>0.201</b> ( $2 \times 10^{-3}$ ) |

Table 2: WSI performance with different representation types. The performance is measured using ARI. Results are presented for all words in the dataset, as well as for verbs and nouns individually. The mean results over 5 runs are given with standard deviation in brackets. The scores that surpass the BERT representations are in bold.

words. The evaluation metric used is the *Adjusted Rand Index* (ARI) (Hubert and Arabie, 1985). ARI measures the similarity between two clusterings by counting the pairs that are assigned to the same or different clusters in both the gold clusters and predicted clusters. It is adjusted to account for chance agreement and gives a score between -1 and 1 where 1 indicates perfect agreement between the two clusterings, while scores below 0 suggest that the match is worse than random chance. For the ARI formula and different clustering evaluation metrics see Appendix C.

We compare the WSI performance with 3 different types of representations. We run the clustering algorithm 5 times for each type of representation with different random states. We report the mean of 5 runs. We apply unpaired t-test to determine if the performance difference is statistically significant. We compare the overall performance and the performance based on grammatical category (verbs and nouns).

## 4.3 Results

Results are shown in Table 2. WSI is performed better with feature-invariant representations than with the original BERT representations for both feature types, with statistically significant differences observed through unpaired t-tests (p-value: 0.0001). The best results are obtained with morphology-invariant representations overall. The largest gain in the performance happens for verbs with morphology-invariant representations. For a more detailed evaluation of the clustering performance using different metrics see Appendix C. Further analysis of the specific cases and reasons behind performance increases and decreases are addressed in the following section.

|               | Syntax |       |       |       |             |             | Morphology |       |       |       |             |             |
|---------------|--------|-------|-------|-------|-------------|-------------|------------|-------|-------|-------|-------------|-------------|
|               | #      | $T_L$ | $T_U$ | Sense | BERT        | Invariant   | #          | $T_L$ | $T_U$ | Sense | BERT        | Invariant   |
| MI            | -      | 9.4   | 27.8  | 14.7  | 26.4        | 23.1        | -          | 4.7   | 40.5  | 9.4   | 41.8        | 25.4        |
| <i>Case 1</i> | 59     | -     | -     | -     | <b>0.34</b> | 0.33        | 11         | -     | -     | -     | <b>0.66</b> | 0.65        |
| <i>Case 2</i> | 34     | -     | -     | -     | 0.07        | <b>0.10</b> | 80         | -     | -     | -     | 0.04        | <b>0.09</b> |
| <i>Case 3</i> | 53     | -     | -     | -     | <b>0.19</b> | 0.18        | 55         | -     | -     | -     | 0.28        | <b>0.29</b> |

Table 3: Relation between the feature types and the WSI performance. MI scores for the association between linguistic features and sense or cluster assignments are given.  $T_L$  and  $T_U$  refers to the lower and upper threshold for MI. For each *Case*, ARI scores for BERT clusters and the clusters formed by the feature-invariant representations are given. Performance increases are in bold.

#### 4.4 Analysis of the Relation Between the Information Types and WSI Performance

Our aim is to determine in which cases the erasure of these information types helps the WSI process. More specifically, we aim to investigate whether, for individual words, word senses are distinguishable by the word’s morphological and syntactic features and therefore, whether the existence or the erasure of the related information helps the WSI process. Even though the overall WSI performance improves with feature-invariant representations, it is possible that for some words, the information erased is actually useful for sense identification. In these cases, the information erasure would negatively affect the WSI process.

In order to investigate this, we identify the three following cases and assess the performance with the original BERT representations and feature-invariant representations for each case:

- *Case 1*: The senses of a word are distinguishable by the word’s morphological, or syntactic features. In this case, we expect the performance with invariant representations to be lower than with the original BERT representations.
- *Case 2*: The senses of a word are *not* distinguishable by the word’s morphological, or syntactic features, but clusters of BERT representations are distinguishable by these features — which then can be assumed to be noise for clustering-based WSI. In this case, we expect the performance with invariant representations to be higher than with the original BERT representations.
- *Case 3*: The senses of a word are *not* distinguishable by the morphological, or syntactic features of the word, and clusters of BERT representations are also *not* distinguishable

by these features. In this case, we expect the performance with invariant representations to be the same as with the original BERT representations.

##### 4.4.1 Method

We measure the association between the features of the word instances and their sense or cluster assignments using *Mutual Information* (MI).<sup>9</sup> We use this information to automatically categorize words into the three cases outlined above.

In order to determine the features of the word instances, we use again the POS tags and dependency labels obtained from spaCy.<sup>10</sup> We refer to them as *linguistic labels*. For each word and for each type of feature we compute three MI scores. Firstly, we calculate the MI score between the linguistic labels of the instances and their gold WordNet sense labels (*Sense MI*). Secondly, we calculate the MI score between the linguistic labels of the instances and their cluster labels, considering the clusters formed by the BERT representations (*BERT MI*). Lastly, we assess the MI score between the linguistic labels of the instances and their cluster labels, considering this time the clusters formed by the feature-invariant representations (*Invariant MI*).

We compare the MI scores to lower ( $T_L$ ) and upper thresholds ( $T_U$ ). The lower and upper threshold are calculated for each feature type as the first quartile and third quartile for all MI scores for this feature. We interpret scores below the lower threshold as indicating no association, and scores above

<sup>9</sup>The mutual information between two variables  $X$  and  $Y$  is defined as follows:

$$MI(X; Y) = \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x, y) \log \left( \frac{p(x, y)}{p(x)p(y)} \right) \quad (2)$$

<sup>10</sup>For some instances of words, the target word is not found in the sentence due to a lemmatization error. We discard these words and experiment with 540 words in total (out of 567).

the upper threshold as indicating an association. We then automatically categorise word types. *Case 1* words have high Sense MI scores, *Case 2* words have low Sense MI and high BERT MI scores, and finally, *Case 3* words have low Sense MI and low BERT MI scores. We compute the average ARI score for each word within each case and compare their performances.

#### 4.4.2 Results

The results of the analysis can be seen in Table 3. MI scores show that the linguistic labels are more strongly associated with BERT clusters than with the sense groups. This suggests that these features are dominant in the BERT clusters more than necessary. This and the fact that ARI performance is lowest for the *Case 2* words indicate that these features introduce noise that affects WSI performance negatively. With the feature-invariant representations, this effect is limited to some extent.

Regarding different *Cases*, the results mostly align with our expectations. Clustering performance with *Case 1* words is slightly higher with original BERT representations. Clustering performance with *Case 2* words is increased with feature-invariant representations. However, the increase for *Case 2* words is much higher than the decrease for *Case 1* words, showing that the erasure of syntactic and morphological information benefits the WSI performance overall. Finally, with *Case 3* words, there is a slight increase or decrease in performance depending on the different feature types.

Regarding different feature types, we observe that the morphological features of words introduce a lot of noise to WSI performance (Sense MI vs. BERT MI). Only for 11 words (out of 540), morphological features of words are found to be associated with different senses (*Case 1*). For 80 words, these features are found to be associated with different BERT clusters, even though they are not relevant to different senses (*Case 2*), therefore introducing noise. Similarly, the average BERT MI score for morphological features surpasses the upper threshold ( $T_U$ ) of association, showing that there is a high level of association between morphological features and BERT clusters. Conversely, syntactic features of words have more relevance to word senses. For 59 words, these features show associations with different senses, and both the Sense MI score is higher, and the difference between Sense and BERT MI scores is lower for this feature type. These differences are also evident

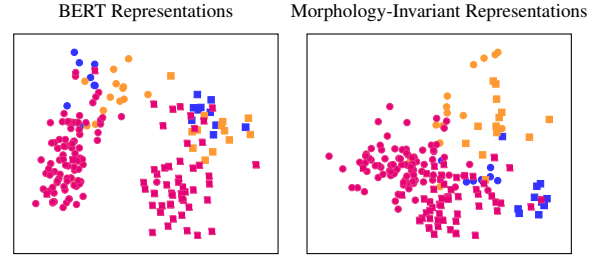


Figure 2: PCA visualizations of BERT representations (left) and morphology-invariant representations (right) of different sense instances of ‘area’. Different data point colors refer to different senses. Different marker styles refer to instances with different morphological features, i.e. grammatical number; *circles* for *singular nouns*, *squares* for *plural nouns*.

in the WSI performance. Erasing morphological information benefits WSI performance more than erasing syntactic information (Table 2).

Let us illustrate these findings with a few examples. The noun ‘area’ has 3 senses in the data. WSI performs worse with BERT representations than with morphology-invariant representations on this word (BERT ARI: 0.03, INV ARI: 0.49). With BERT representations, we observe that two clusters are formed and that they are formed mostly based on the grammatical number of the instances, although there is no association between grammatical number and the senses of the word (Sense MI: 0.0, BERT MI: 57.7, Invariant MI: 4.2). With morphology-invariant representations, we observe that this pattern is broken. Grammatical number does not affect the similarity of the representations and the instances of each sense are closer to each other. Singular and plural instances of the third sense are successfully clustered together. Even though first and second sense instances form only one cluster, instances of each sense are closer to each other and the senses are more separable. See Figure 2 for the PCA visualization of the different representations of ‘area’ instances. For a more detailed plot see Figure 4 in Appendix D.

Let’s consider a contrasting example. The noun ‘field’ has 4 senses in the data. WSI performs better with BERT representations than syntax-invariant representations (BERT ARI: 0.62, INV ARI: 0.26) and senses are associated with the syntactic features of the word (Sense MI: 40.3, BERT MI: 43.7, Invariant MI: 11.0). The 3rd sense of ‘field’ has the meaning ‘somewhere (away from a studio or office or library or laboratory) where practical work is done or data is collected’ and almost all of its

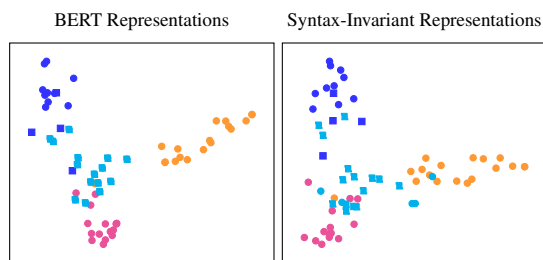


Figure 3: PCA visualizations of BERT representations (left) and syntax-invariant representations (right) of different sense instances of ‘field’. Different data point colors refer to different senses. *Light blue* data points represent the 3rd sense instances of the noun. Different marker styles refer to instances with different syntactic roles; *squares* for *compounds heads*, *circles* for others.

instances are heads of a compound as in example (2).

- (2)
  - a. They will give suggestions that can be worked up into **field** procedures.
  - b. Actually, none of these papers says much directly about **field** techniques.

The 3rd sense instances are clustered together when BERT representations are used. After the erasure of syntactic information, their representations are closer to the representations of other sense instances. As a result, they are clustered with other sense instances when syntax-invariant representations are used, resulting in poor performance. See Figure 3 for the PCA visualization of the different representations of ‘field’ instances. For a more detailed plot see Figure 5 in Appendix D.

## 5 Conclusion

We adapt the framework proposed by Jaiswal et al. (2020) in order to erase specific information from the representations of LMs. With this method, we create two types of representations from BERT embeddings: invariant to either (i) morphological features or (ii) syntactic features. Our results show that the resulting feature-invariant representations are more suitable for the WSI task. Furthermore, we show that even though some syntactic features provide valuable information for WSI, both types of features introduce noise that, overall, negatively impacts the performance of clustering-based WSI.

## 6 Acknowledgement

This study is funded by Labex EFL and the DFG project “Coercion and Copredication as Flexible

Frame Composition”. We would like to thank the anonymous reviewers for their valuable comments.

## References

- Enrique Amigó, Julio Gonzalo, Javier Artiles, and Felisa Verdejo. 2009. A comparison of extrinsic clustering evaluation metrics based on formal constraints. *Information retrieval*, 12:461–486.
- Amit Bagga and Breck Baldwin. 1998. Entity-based cross-document coreferencing using the vector space model. In *COLING 1998 Volume 1: The 17th International Conference on Computational Linguistics*.
- Alexander M Bronstein, Michael M Bronstein, and Ron Kimmel. 2003. Expression-invariant 3d face recognition. In *international conference on Audio-and video-based biometric person authentication*, pages 62–70. Springer.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [BERT: Pre-training of deep bidirectional transformers for language understanding](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
- Christiane Fellbaum. 1998. [WordNet: An electronic lexical database](#). MIT press.
- Christian Hadiwinoto, Hwee Tou Ng, and Wee Chung Gan. 2019. [Improved word sense disambiguation using pre-trained contextualized word representations](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5297–5306, Hong Kong, China. Association for Computational Linguistics.
- Lawrence Hubert and Phipps Arabie. 1985. Comparing partitions. *Journal of classification*, 2:193–218.
- Ayush Jaiswal, Daniel Moyer, Greg Ver Steeg, Wael AbdAlmageed, and Premkumar Natarajan. 2020. Invariant representations through adversarial forgetting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 4272–4279.
- Ayush Jaiswal, Rex Yue Wu, Wael Abd-Almageed, and Prem Natarajan. 2018. Unsupervised adversarial invariance. *Advances in neural information processing systems*, 31.
- Alireza Khotanzad and Yaw Hua Hong. 1990. Rotation invariant image recognition using features selected via a systematic method. *Pattern recognition*, 23(10):1089–1101.

- Henry Kučera, Winthrop Francis, William Freeman Twaddell, Mary Lois Marckworth, Laura M Bell, and John Bissell Carroll. 1967. Computational analysis of present-day american english. *Brown University Press, Providence, RI*.
- Severin Laicher, Sinan Kurtiyigit, Dominik Schlechtweg, Jonas Kuhn, and Sabine Schulte im Walde. 2021. [Explaining and improving BERT performance on lexical semantic change detection](#). In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Student Research Workshop*, pages 192–202, Online. Association for Computational Linguistics.
- Christos Louizos, Kevin Swersky, Yujia Li, Max Welling, and Richard Zemel. 2017. The variational fair autoencoder.
- Daniel Loureiro, Kiamehr Rezaee, Mohammad Taher Pilehvar, and Jose Camacho-Collados. 2021. [Analysis and evaluation of language models for word sense disambiguation](#). *Computational Linguistics*, 47(2):387–443.
- Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. [Building a large annotated corpus of English: The Penn Treebank](#). *Computational Linguistics*, 19(2):313–330.
- George A. Miller, Claudia Leacock, Randee Teng, and Ross T. Bunker. 1993. [A semantic concordance](#). In *Human Language Technology: Proceedings of a Workshop Held at Plainsboro, New Jersey, March 21-24, 1993*.
- Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. 2011. [Scikit-learn: Machine learning in python](#). *Journal of Machine Learning Research*, 12(85):2825–2830.
- Minlong Peng and Qi Zhang. 2020. [Weighed domain-invariant representation learning for cross-domain sentiment analysis](#). In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 251–265, Barcelona, Spain (Online). International Committee on Computational Linguistics.
- Mohammad Taher Pilehvar and Jose Camacho-Collados. 2019. [WiC: the word-in-context dataset for evaluating context-sensitive meaning representations](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1267–1273, Minneapolis, Minnesota. Association for Computational Linguistics.
- Andrew Rosenberg and Julia Hirschberg. 2007. V-measure: A conditional entropy-based external cluster evaluation measure. In *Proceedings of the 2007 joint conference on empirical methods in natural language processing and computational natural language learning (EMNLP-CoNLL)*, pages 410–420.
- Hassan Sajjad, Nadir Durrani, Fahim Dalvi, Firoj Alam, Abdul Khan, and Jia Xu. 2022. [Analyzing encoded concepts in transformer language models](#). In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3082–3101, Seattle, United States. Association for Computational Linguistics.
- Younes Samih and Laura Kallmeyer. 2023. [Unsuper-vised semantic frame induction revisited](#). In *Proceedings of the 15th International Conference on Computational Semantics*, pages 89–93, Nancy, France. Association for Computational Linguistics.
- Pierre-Yves Vandenbussche, Tony Scerri, and Ron Daniel Jr. 2021. [Word sense disambiguation with transformer models](#). In *Proceedings of the 6th Workshop on Semantic Deep Learning (SemDeep-6)*, pages 7–12, Online. Association for Computational Linguistics.
- Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. [Transformers: State-of-the-art natural language processing](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.
- Qizhe Xie, Zihang Dai, Yulun Du, Eduard Hovy, and Graham Neubig. 2017. Controllable invariance through adversarial feature learning. *Advances in neural information processing systems*, 30.
- Ji Xin, Chenyan Xiong, Ashwin Srinivasan, Ankita Sharma, Damien Jose, and Paul Bennett. 2022. [Zero-shot dense retrieval with momentum adversarial domain invariant representations](#). In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 4008–4020, Dublin, Ireland. Association for Computational Linguistics.
- Deniz Ekin Yavas. 2024. [Assessing the significance of encoded information in contextualized representations to word sense disambiguation](#). In *Proceedings of the Third Workshop on Understanding Implicit and Underspecified Language*, pages 42–53, Malta. Association for Computational Linguistics.
- Ying Zhao and George Karypis. 2001. Criterion functions for document clustering: Experiments and analysis.

## A Model Architectures

- **Discriminator for Syntactic Information:** 2-layer nonlinear neural network for classification. ReLU is used as the activation function. Given a token embedding, it predicts the dependency label of the token. Output dimension is the number of classes. Cross-entropy loss is used as the loss function and Adam is used as the optimizer.
- **Discriminator for Morphological Information:** 2-layer nonlinear neural network for classification. ReLU is used as the activation function. Given a token embedding, it predicts the fine-grained POS tag of the token. Output dimension is the number of classes found in the dataset. Cross-entropy loss is used as the loss function and Adam is used as the optimizer.
- **Predictor:** 3-layer linear neural network that maps the token embedding to the vocabulary of BERT with size 30522. 2nd layer is for normalization and drop-out (0.1) is applied before the output layer. Cross-entropy loss is used as the loss function and Adam is used as the optimizer.
- **Forget-Gate:** 3-layer nonlinear neural network that transforms the input embedding. ReLU is used as the activation function. Cross-entropy loss is used as the loss function and Adam is used as the optimizer.

## B Model Training Procedures

**Adversarial Models.** All adversarial models are trained with batch size 128 and learning rate  $10^{-6}$  for the predictor,  $10^{-5}$  for the discriminator, and  $10^{-4}$  for the forget gate. All the models are trained for 800 epochs.

### Upper Bounds.

- **Probing for syntactic information:** 74 epochs, batch size 128, learning rate  $10^{-5}$ .
- **Probing for morphological information:** 66 epochs, batch size 128, learning rate  $10^{-5}$ .
- **Word prediction:** 132 epochs, batch size 128, learning rate  $10^{-6}$ .

### Lower Bounds.

- **Probing for syntactic information:** 9 epochs, batch size 128, learning rate  $10^{-4}$ .

## Feature-Invariant Representations Evaluation.

- **Morphology-Invariant Representations:** The word predictor is trained for 30 epochs with batch size 128, learning rate  $10^{-5}$ . The probing model is trained for 22 epochs with batch size 128, learning rate  $10^{-4}$ . The best forget-gate is obtained from the 400th epoch of the adversarial model’s training.
- **Syntax-Invariant Representations:** The word predictor is trained for 20 epochs with batch size 128, learning rate  $10^{-5}$ . The probing model is trained for 37 epochs with batch size 128, learning rate  $10^{-4}$ . The best forget-gate is obtained from the 500th epoch of the adversarial model’s training.

## C Clustering Performance Details

We evaluate the clustering performance using metrics from 4 different categories based on the categorization in Amigó et al. (2009) because different categories have different strengths in measuring clustering quality; metrics based on set matching (*Purity*, *Inverse Purity* (Zhao and Karypis, 2001) and their harmonic mean *PIF*), metrics based on entropy (*V-Measure* (Rosenberg and Hirschberg, 2007)), metrics based on counting pairs (*Adjusted Rand Index* (Hubert and Arabie, 1985)), and BCubed metrics (*BCubed Precision*, *Recall* and *F-score* (Bagga and Baldwin, 1998)).

C being the set of clusters and L being the true grouping, *Purity* and *Inverse Purity* are calculated as follows:

$$\text{Purity} = \sum_i \frac{|C_i|}{N} \max_j \text{Precision}(C_i, L_j) \quad (3)$$

$$\text{Inverse Purity} = \sum_i \frac{|L_i|}{N} \max_j \text{Precision}(L_i, C_j) \quad (4)$$

*V-measure* is calculated based on the homogeneity and completeness. Homogeneity measures how much each cluster contains only data points that are members of a single class. Completeness measures how much all data points that are members of a given class are assigned to the same cluster. It is calculated as follows:

|                 | ARI            | V-M            | PU             | IPU            | PIF            | P-BCubed       | R-BCubed       | F-BCubed       |
|-----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|----------------|
| BERT            | 0.210 (0.0008) | 0.265 (0.0007) | 0.732 (0.0007) | 0.658 (0.001)  | 0.670 (0.001)  | 0.652 (0.0006) | 0.575 (0.0008) | 0.580 (0.0004) |
| Syn-Invariant   | 0.221 (0.001)  | 0.274 (0.001)  | 0.732 (0.0006) | 0.682 (0.0009) | 0.684 (0.0008) | 0.653 (0.0007) | 0.599 (0.001)  | 0.594 (0.0008) |
| Morph-Invariant | 0.232 (0.001)  | 0.283 (0.0006) | 0.736 (0.0004) | 0.683 (0.001)  | 0.688 (0.0007) | 0.657 (0.0004) | 0.598 (0.0008) | 0.597 (0.0005) |

Table 4: Clustering evaluation results with different representations with different metrics for all words in the data. The mean results over 5 runs are given with standard deviation in brackets.

|                 | ARI            | V-M            | PU             | IPU           | PIF           | P-BCubed       | R-BCubed      | F-BCubed       |
|-----------------|----------------|----------------|----------------|---------------|---------------|----------------|---------------|----------------|
| BERT            | 0.251 (0.001)  | 0.309 (0.001)  | 0.769 (0.0007) | 0.689 (0.002) | 0.707 (0.001) | 0.699 (0.0004) | 0.607 (0.001) | 0.623 (0.0008) |
| Syn-Invariant   | 0.263 (0.0004) | 0.320 (0.0005) | 0.772 (0.0006) | 0.682 (0.001) | 0.706 (0.001) | 0.703 (0.0003) | 0.600 (0.001) | 0.622 (0.0009) |
| Morph-Invariant | 0.267 (0.001)  | 0.322 (0.001)  | 0.772 (0.001)  | 0.682 (0.001) | 0.705 (0.001) | 0.704 (0.001)  | 0.598 (0.001) | 0.620 (0.001)  |

Table 5: Clustering evaluation results with different representations with different metrics for nouns. The mean results over 5 runs are given with standard deviation in brackets.

|                 | ARI           | V-M           | PU            | IPU           | PIF            | P-BCubed      | R-BCubed       | F-BCubed       |
|-----------------|---------------|---------------|---------------|---------------|----------------|---------------|----------------|----------------|
| BERT            | 0.174 (0.001) | 0.227 (0.001) | 0.699 (0.001) | 0.632 (0.001) | 0.638 (0.0008) | 0.611 (0.001) | 0.547 (0.0006) | 0.542 (0.0001) |
| Syn-Invariant   | 0.185 (0.002) | 0.233 (0.002) | 0.698 (0.001) | 0.681 (0.002) | 0.665 (0.001)  | 0.608 (0.001) | 0.598 (0.001)  | 0.570 (0.001)  |
| Morph-Invariant | 0.201 (0.002) | 0.248 (0.002) | 0.705 (0.001) | 0.684 (0.002) | 0.672 (0.002)  | 0.616 (0.001) | 0.599 (0.002)  | 0.576 (0.002)  |

Table 6: Clustering evaluation results with different representations with different metrics for verbs. The mean results over 5 runs are given with standard deviation in brackets.

$$V = 2 \times \frac{\text{Homogeneity} \times \text{Completeness}}{\text{Homogeneity} + \text{Completeness}} \quad (5)$$

*Adjusted Rand Index* (ARI) adjusts the *Rand Index* (RI) to account for chance agreement. *RI* calculates the similarity between two clusterings by considering pairs of samples and determining whether they are assigned to the same cluster or different clusters in both clusterings. They are calculated as follows:

$$RI = \frac{\text{correct similar pairs} + \text{correct dissimilar pairs}}{\text{total number of pairs}} \quad (6)$$

$$ARI = \frac{\max(RI) - \text{Expected\_RI}}{RI - \text{Expected\_RI}} \quad (7)$$

*Correctness* is the relation between  $e$  and  $e'$  in the distribution, where  $C(e)$  denotes the cluster and  $L(e)$  true grouping of the item. Correctness means that both items have the same category and belong to the same cluster. The overall *Precision BCubed* and *Recall BCubed* are obtained by averaging the precision and recall scores of all items in the dataset as follows:

$$\text{Precision BCubed} = \text{Avg}_e [\text{Avg}_{e' \cdot C(e)=C(e')} [\text{Correctness}(e, e')]] \quad (8)$$

$$\text{Recall BCubed} = \text{Avg}_e [\text{Avg}_{e' \cdot L(e)=L(e')} [\text{Correctness}(e, e')]] \quad (9)$$

The detailed evaluation of the clustering performance with different metrics for all words can be seen in Table 4, for nouns in Table 5 and verbs in Table 6. The mean results over 5 runs are given.

## D Clustering Visualizations

The PCA visualizations of the BERT representations and morphology-invariant representations of ‘area’ instances can be seen in Figure 4. Similarly, the PCA visualizations of the BERT representations and syntax-invariant representations of ‘field’ instances can be seen in Figure 5.

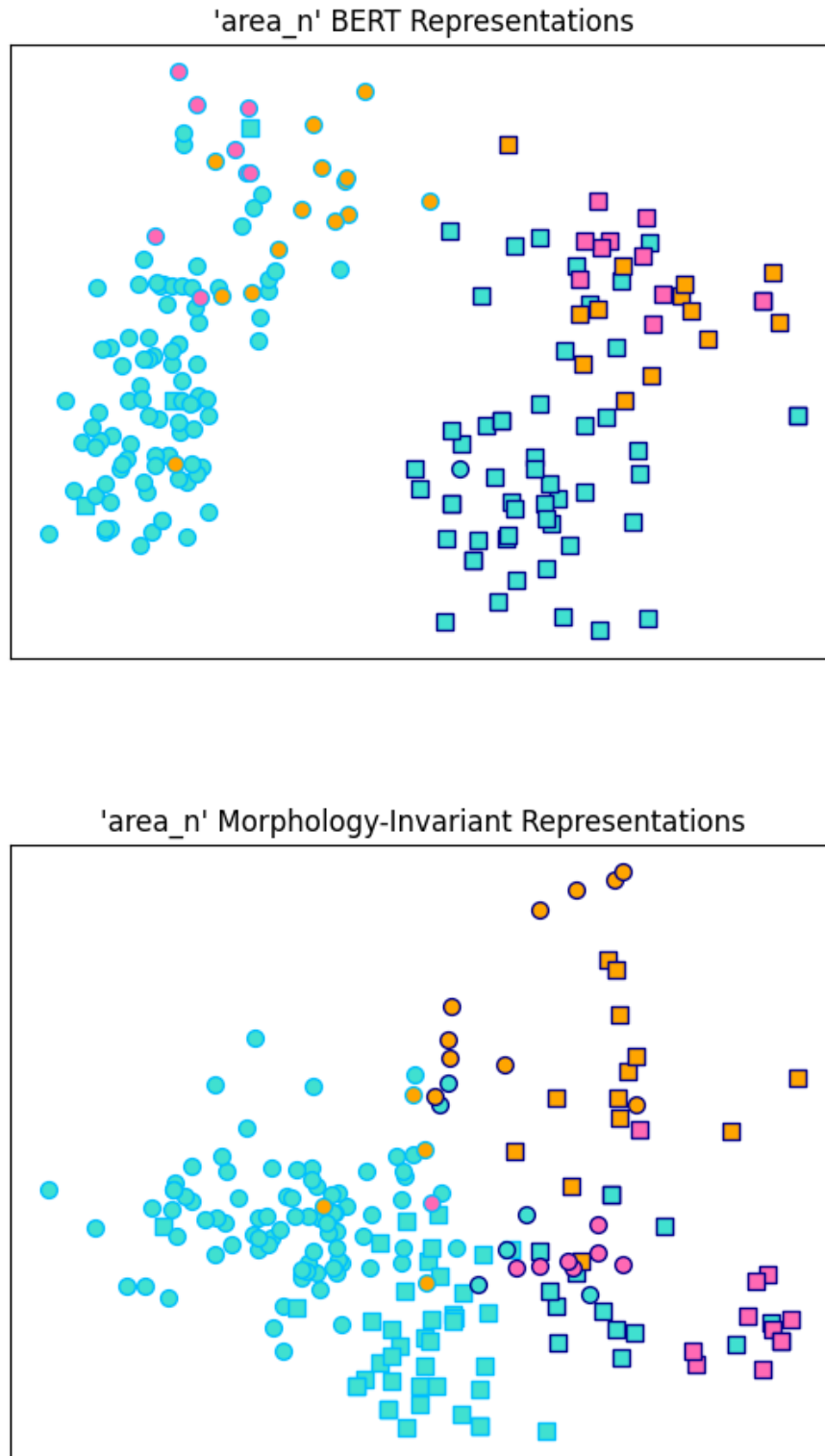


Figure 4: PCA visualizations of BERT representations (top) and morphology-invariant representations (bottom) of different sense instances of ‘area’. Different data point colors refer to different senses, and different border colors refer to different clusters. Additionally, different marker styles refer to instances with different morphological features, i.e. grammatical number; *circles* for *singular nouns*, *squares* for *plural nouns*.

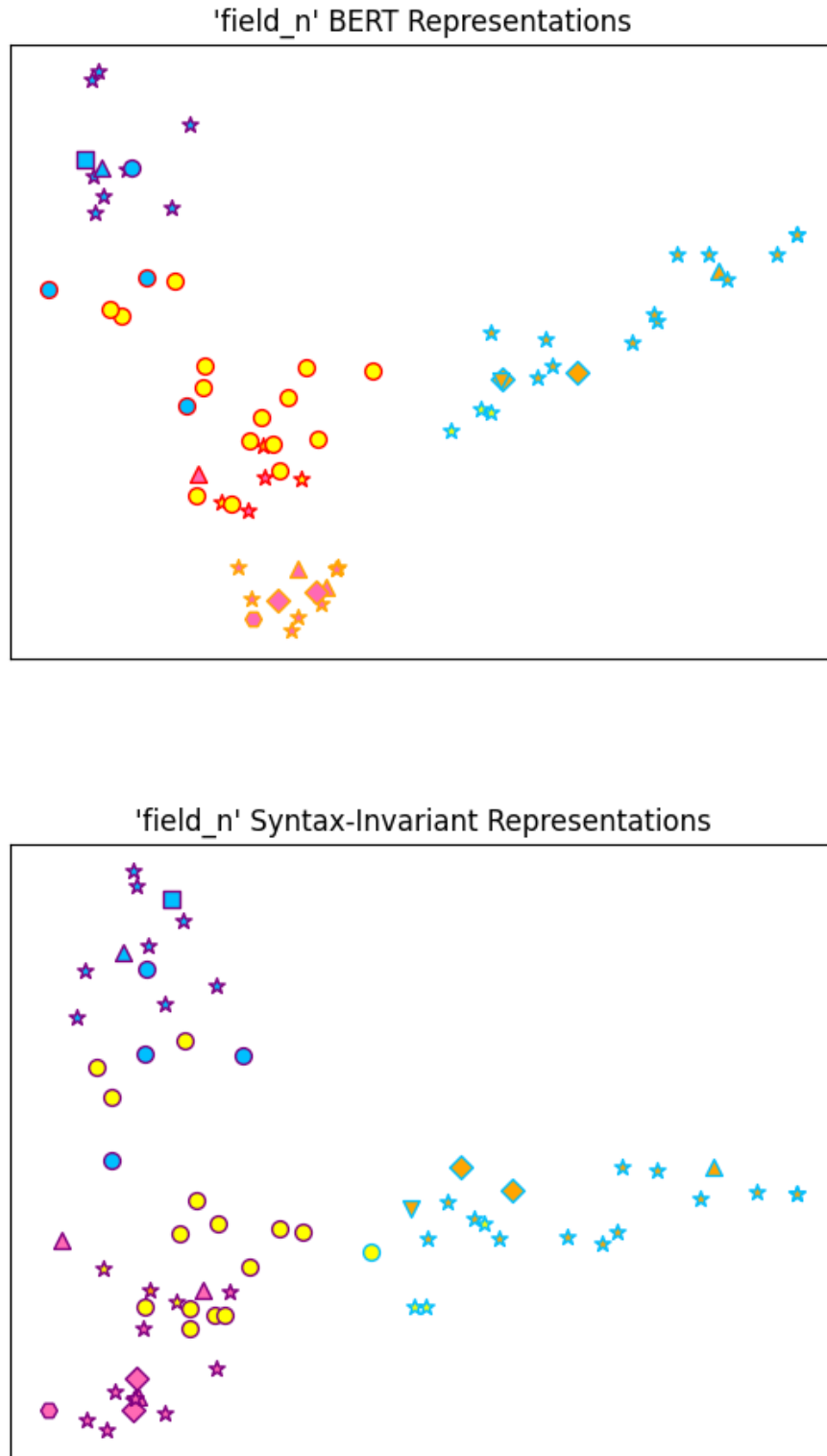


Figure 5: PCA visualizations of BERT representations (top) and syntax-invariant representations (bottom) of different sense instances of ‘field’. Different data point colors refer to different senses, and different border colors refer to different clusters. Additionally, different marker styles refer to instances with different syntactic roles; *circles* for *compound heads*, *stars* for *prepositional objects*, *triangles* for *direct objects*, *diamonds* for *subjects*, *reversed triangles* for *passive subjects*, and *hexagons* for *attributes*.



## Chapter 6

# On the Relation Between Fine-Tuning, Topological Properties, and Task Performance in Sense-Enhanced Embeddings

# On the Relation Between Fine-Tuning, Topological Properties, and Task Performance in Sense-Enhanced Embeddings

Deniz Ekin Yavas<sup>1</sup>, Timothée Bernard<sup>2</sup>, Benoît Crabbé<sup>2</sup>, Laura Kallmeyer<sup>1</sup>

Heinrich Heine University Düsseldorf<sup>1</sup>, Université Paris Cité<sup>2</sup>

{deniz.yavas, laura.kallmeyer}@hhu.de<sup>1</sup>

{timothee.bernard, benoit.crabbe}@u-paris.fr<sup>2</sup>

## Abstract

Topological properties of embeddings, such as isotropy and uniformity, are closely linked to their expressiveness, and improving these properties enhances the embeddings' ability to capture nuanced semantic distinctions. However, fine-tuning can reduce the expressiveness of the embeddings of language models. This study investigates the relation between fine-tuning, topology of the embedding space, and task performance in the context of sense knowledge enhancement, focusing on identifying the topological properties that contribute to the success of sense-enhanced embeddings. We experiment with two fine-tuning methods: *Supervised Contrastive Learning (SCL)* and *Supervised Predictive Learning (SPL)*. Our results show that SPL, the most standard approach, exhibits varying effectiveness depending on the language model and is inconsistent in producing successful sense-enhanced embeddings. In contrast, SCL achieves this consistently. Furthermore, while the embeddings with only increased *sense-alignment* show reduced task performance, those that also exhibit high *isotropy* and balance *uniformity* with *sense-alignment* achieve the best results. Additionally, our findings indicate that supervised and unsupervised tasks benefit from these topological properties to varying degrees.

## 1 Introduction

The contextualized word embeddings of pre-trained language models (LMs) are powerful tools, but do not align well with word senses: the embedding spaces typically do not exhibit distinct clusters corresponding to similar meaning (Yenicelek et al., 2020). Therefore, using these embeddings as such does not always yield satisfactory results on word sense identification tasks (Pilehvar and Camacho-Collados, 2019; Samih and Kallmeyer, 2023; Yavas, 2024; Yavas et al., 2024). One way to overcome this problem is to enhance the embeddings with external sense knowledge (Peters et al.,

2019; Lauscher et al., 2020; Garí Soler and Apidianaki, 2020; Levine et al., 2020; Mosolova et al., 2024).

Fine-tuning is a widely adopted method for *knowledge enhancement*, where LM embeddings are updated by incorporating external knowledge.<sup>1</sup> However, fine-tuning can also reduce the expressiveness of the LM embeddings, i.e., their ability to capture and distinguish nuanced relations, due to reduced isotropy (Zhang et al., 2022). Therefore, it is essential to determine the effects of fine-tuning on the topology of the embedding space in the context of sense knowledge enhancement. Furthermore, it is also important to understand the relation between topological properties and task performance. Despite their importance, these relations remain unexplored, and this study aims to fill this gap. To this end, we compare different fine-tuning methods, investigate how they transform the embedding space of LMs, evaluate the resulting embeddings in terms of task performance, and finally, identify the topological properties associated with successful sense-enhanced embeddings.

We evaluate the embeddings based on three topological properties: *sense-alignment*, *uniformity* (Wang and Isola, 2020), and *isotropy* (Mu and Viswanath, 2018). *Sense-alignment* evaluates how close to each other are the embeddings of word occurrences sharing the same sense. *Isotropy* evaluates how evenly the embeddings are distributed across all directions in the embedding space, while *uniformity* measures the density of their distribution, evaluating whether they spread uniformly in the embedding space.

Both isotropy and uniformity are related to the expressiveness of the embeddings (Gao et al., 2021; Rajaei and Pilehvar, 2021a; Zhang et al., 2022). Ideally, an embedding space should be uniform and isotropic, while maintaining semantic relations

<sup>1</sup>See the survey by Hu et al. (2023).

as much as possible. This enables utilization of the entire space and directions to capture nuanced semantic relations and differences between word occurrences. However, LM embeddings have inherently limited expressiveness (Gao et al., 2019). As a result, even the embeddings of random words often exhibit high similarity (Ethayarajh, 2019). However, improving the uniformity and isotropy of the LM embeddings can lead to better performance on semantics-related tasks (Biś et al., 2021; Gao et al., 2021; Liu et al., 2021a,b; Rajaei and Pilehvar, 2021a).

We experiment with two fine-tuning methods: *Supervised Contrastive Learning (SCL)* and *Supervised Predictive Learning (SPL)*. They differ primarily in their objectives: SCL aims to push instances of the same class (i.e., sense) together while pushing instances from different classes apart; SPL aims to predict the correct class of an instance. We are particularly interested in these methods because SPL is the standard fine-tuning method for LMs, while fine-tuning via Contrastive Learning has been shown to improve the isotropy and uniformity of the embedding space (see Gao et al., 2021; Liu et al., 2021a,b; Yan et al., 2021, which, however, do not focus on knowledge enhancement).<sup>2</sup>

The few studies that compare SPL and SCL have primarily focused on performance differences, showing better performance with SCL in different contexts, but without investigating the quality of the resulting embeddings (Gunel et al., 2020; Khosla et al., 2020). In the context of sense knowledge enhancement, we aim to investigate whether the performance advantage of SCL holds and to understand the underlying factors in terms of topological properties in performance differences. For this purpose, we conduct detailed experiments, including experiments focusing on the temperature parameter of the loss functions used, as this parameter influences class separation and has been shown to impact the topological properties of the embeddings in Unsupervised Contrastive Learning (UCL; Wang and Liu, 2021).

We enhance the embeddings of LMs with WordNet supersense information (Fellbaum, 1998) using different fine-tuning methods, and evaluate the word sense identification performance with the resulting embeddings on the Word-in-Context task (WiC; Pilehvar and Camacho-Collados, 2019).

<sup>2</sup>These studies mostly focus on Unsupervised Contrastive Learning, a related but distinct concept that does not rely on explicit external knowledge.

WiC is a binary classification task, where the goal is to determine if a target word form (e.g., *bass*) has the same meaning in two different contexts (e.g., *she plays the bass* and *this fish is a bass*). We experiment with both supervised and unsupervised setups for the task. This allows us to investigate whether supervised and unsupervised tasks benefit from different topological properties in embeddings to achieve good performance.

In addition to these fine-tuning methods, we conduct further experiments to investigate the effects of two factors on task performance: *task similarity*, and *isotropy in isolation*. For task similarity, we fine-tune the LM on a task similar to the downstream task (WiC). We refer to this fine-tuning approach as *task adaptation*. For isotropy in isolation, we apply the isotropization post-processing method proposed by Mu and Viswanath (2018) to the original LM embeddings, allowing us to investigate the effects of isotropy without incorporating external knowledge to the embeddings. We further evaluate the embeddings obtained from both methods.

Our study provides novel insights into the relation between fine-tuning, topology of the embedding space, and task performance in the context of sense knowledge enhancement. The key findings of this work are as follows.

- *SPL* and *SCL* fine-tuning methods transform the embedding spaces of the LMs differently with respect to sense-alignment, uniformity, and isotropy.
- The properties of the embeddings obtained after fine-tuning appear consistently, regardless of the original embeddings.
- The effectiveness of *SPL* shows mixed results, varying depending on the LM. The embeddings created via *SPL* do not outperform the original LM embeddings in some cases. In contrast, *SCL* shows more consistent improvements and often achieves the highest scores, making it a more effective method for sense knowledge enhancement.
- The embeddings with only increased *sense-alignment* perform worse than the original LM embeddings. The highest performance is achieved with those that exhibit a significant increase in both *isotropy* and *sense-alignment*.
- The embeddings that achieve the best performances overall balance uniformity and sense-

alignment.

- The relative importance of these properties varies across supervised and unsupervised setups.

## 2 Related Work

### 2.1 Enhancing LM Embeddings with Sense Knowledge

Prior research on sense knowledge enhancement has primarily focused on injecting external knowledge during the pre-training phase of the LMs (Peters et al., 2019; Lauscher et al., 2020; Levine et al., 2020). A few studies have fine-tuned LMs for a similar purpose (Garí Soler and Apidianaki, 2020; Mosolova et al., 2024). Garí Soler and Apidianaki (2020) fine-tune BERT on semantic similarity datasets using both a classification head (via SPL) and cosine distance head, which utilizes Cosine Embedding Loss to adjust the embedding distances according to their meaning similarity. They show improved performance on the Graded Word Similarity in Context task (Armendariz et al., 2020) over the original LM embeddings. Mosolova et al. (2024) fine-tune BERT via SCL using examples of use of different senses of the same word form. They show improved performance on the WiC task over the original LM embeddings. Despite the popularity of SPL in other contexts, these studies use other methods than SPL alone. It is still unclear whether SPL can produce similar results, how the two methods differently alter the embedding space, and which topological properties may be associated with any performance differences.

Finally, Bihani and Rayz (2021) introduce a post-processing method to generate isotropic sense-enhanced embeddings. Their approach integrates and adapts the isotropization method of Mu and Viswanath (2018) with the retrofitting method proposed by Faruqui et al. (2015). However, they do not evaluate the resulting embeddings in terms of task performance.

### 2.2 Topology of the Embedding Space

#### 2.2.1 Alignment and Uniformity

Wang and Isola (2020) propose metrics *alignment*<sup>3</sup> and *uniformity*. They compare SPL and UCL for

<sup>3</sup>They propose alignment in the context of UCL. The aim of UCL is usually to push the embeddings of a data instance and of its augmented (as in “data augmentation”) version closer, while pushing the embeddings of other instances apart. Alignment quantifies the closeness between a data point and its augmented version.

pre-training encoders for language and vision and show that the embeddings produced by these methods differ in terms of alignment and uniformity, as well as downstream task performance. They find that UCL produces more uniform embeddings, leading to improved performance. Furthermore, they emphasize the importance of balancing alignment and uniformity to generate successful embeddings in terms of downstream task performance.

A similar effect of uniformity has been observed with BERT embeddings specifically. Gao et al. (2021) focus on improving the sentence embeddings of BERT and show that fine-tuning the model via UCL or SCL results in embeddings exhibiting increased uniformity and leading to better results on Semantic Textual Similarity Tasks.

Finally, Wang and Liu (2021) have shown a connection between the temperature parameter in the loss and both the *uniformity* and *tolerance* of the embeddings in UCL. Tolerance is measured as the average similarity of instances within the same class. In their experiments, successful embeddings are only obtained when balancing these two properties.

#### 2.2.2 Isotropy

Previous studies have observed that the embeddings of LMs are anisotropic, and it has been suggested that isotropy is linked to better performance on semantics-related tasks (Biś et al., 2021; Gao et al., 2021; Liu et al., 2021a,b; Rajaei and Pilehvar, 2021a; Yan et al., 2021). Several methods have been proposed to improve isotropy, including regularization techniques during the pre-training of the LM (Gao et al., 2019; Wang et al., 2019a,b; Zhang et al., 2020) and post-processing methods (Bihani and Rayz, 2021; Biś et al., 2021; Rajaei and Pilehvar, 2021a). In addition, fine-tuning LMs via Contrastive Learning has been shown to increase the isotropy of their embeddings (Gao et al., 2021; Liu et al., 2021a,b; Yan et al., 2021).

Anisotropy is also seen in fine-tuned LM embeddings (Rajaei and Pilehvar, 2021b; Zhang et al., 2022) and fine-tuning has been shown to increase anisotropy (Zhang et al., 2022). Furthermore, post-processing methods designed to increase the isotropy of pre-trained LM embeddings tend to be detrimental in terms of downstream task performance when applied to the embeddings produced by fine-tuned LMs (Rajaei and Pilehvar, 2021b; Zhang et al., 2022).

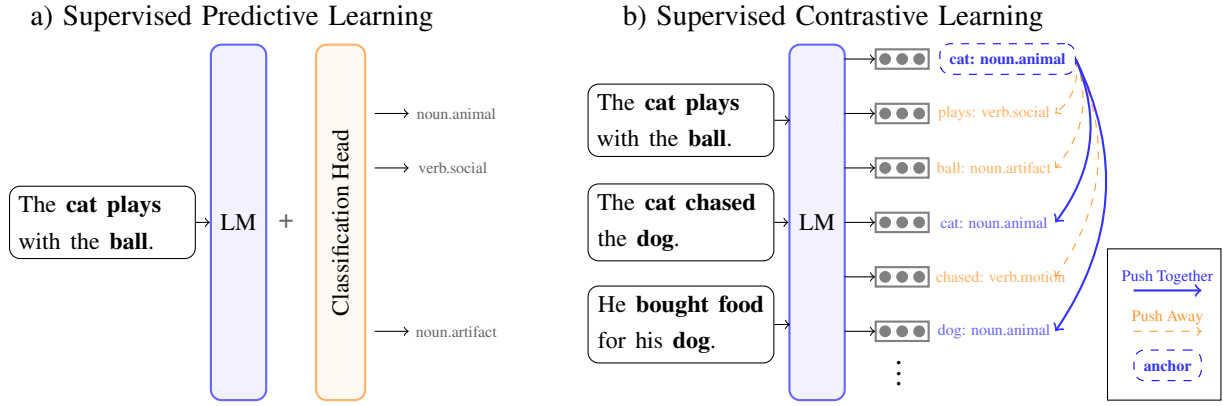


Figure 1: Two fine-tuning methods for sense knowledge enhancement based on WordNet supersenses. For Supervised Predictive Learning (a), a supersense classification head is added to the encoder. For Supervised Contrastive Learning (b), for each *anchor* (all nouns and verbs occurrences in a batch of sentences are simultaneously anchors), the encoder is fine-tuned to push closer to the anchor the word occurrences with the same supersense while pushing away from it those with a different supersense.

### 3 Methodology

In this section, we describe our external source for sense knowledge (Section 3.1) and the three methods that we use to enhance the contextualized word embeddings produced by two LMs: BERT (*base-uncased*) (Devlin et al., 2019) and RoBERTa (*base*) (Liu et al., 2019).

These methods are (1) fine-tuning the LMs for sense knowledge enhancement using SPL and SCL (Section 3.2), (2) fine-tuning the LMs for task adaptation (Section 3.3), and (3) applying an isotropization post-processing method to the original LM embeddings (Section 3.4). The resulting sets of embeddings are evaluated in Section 4. To prevent over-fitting during fine-tuning (first two methods), the first 8 layers of the LMs are frozen<sup>4</sup> and a dropout rate of 0.7 is used after each hidden layer.<sup>5</sup>

#### 3.1 Data

To fine-tune the LMs for both sense knowledge and task adaptation, we use the SemCor corpus (Miller et al., 1993), which is based on WordNet (Fellbaum, 1998). SemCor provides sentences in which word occurrences that belong to certain parts-of-speech are annotated with a WordNet synset label, which in turn is linked to a *supersense*. Supersenses are broad semantic categories that group word senses and are easier to identify than word-specific senses. For example, “bass” maps to the

supersense artifact in “she plays the *bass*” and to animal in “this fish is a *bass*”. Since the WiC task focuses only on verbs and nouns, we ignore all other parts-of-speech. We split the corpus into train, validation, and test sets (70:15:15) and use the test set to evaluate the topology of the embedding spaces, as detailed in Section 4.1.

#### 3.2 Fine-Tuning for Sense Knowledge Enhancement

For Supervised Predictive Learning (SPL), a classification head is added to the LM and the model is fine-tuned by minimization of the Cross-Entropy loss to predict the supersense label of each verb and noun token.<sup>6</sup> See Figure 1, (a), for an illustration.

For Supervised Contrastive Learning (SCL), the LM is fine-tuned by minimization of the SCL loss (Khosla et al., 2020) to push together the embeddings of the word occurrences that have the same supersense label while pushing apart the embeddings of the word occurrences with different supersense labels.<sup>7</sup> See Figure 1, (b), for an illustration. The SCL loss for a batch is given by Formula 1, where  $f$  is a vector similarity metric,  $\tau$  is the temperature parameter<sup>8</sup>,  $I$  is the set of anchors in the batch (all noun and verb occurrences), for each  $i \in I$ ,  $P(i)$  is the set of positive instances for  $i$

<sup>6</sup>As these LMs work with subword tokens, we consider two labels for each supersense; one (B-) used for tokens starting a word occurrence, and one (I-) used for other tokens.

<sup>7</sup>We use as embedding of a word occurrence the average of the embeddings of its subword tokens, as suggested by Mosolova et al. (2024).

<sup>8</sup>The temperature controls the separation between positive and negative instances. See Section 5 for details.

<sup>4</sup>We experimented with freezing either no layer, the first 4 layers, the first 8 layers, or the first 11 layers. Freezing the first 8 layers gave the best WiC task performance.

<sup>5</sup>Code is available at: <https://github.com/yavasde/Fine-Tuning-For-Sense-Enhancement/>

(i.e., word occurrences with the same label as  $i$ , excluding  $i$  itself), and  $A(i)$  is the set of all positive or negative instances for  $i$  in the batch (again, excluding  $i$  itself).

$$\mathcal{L}_{\text{SCL}} = - \sum_{i \in I} \frac{1}{|P(i)|} \sum_{p \in P(i)} \log \left( \frac{\exp \left( \frac{f(\mathbf{z}_i, \mathbf{z}_p)}{\tau} \right)}{\sum_{a \in A(i)} \exp \left( \frac{f(\mathbf{z}_i, \mathbf{z}_a)}{\tau} \right)} \right) \quad (1)$$

The batches are sampled randomly. Following previous studies (Chen et al., 2020; Gao et al., 2021; Yan et al., 2021; Mosolova et al., 2024), we use *Cosine Similarity*.

We experiment with different temperature values for both SCL and SPL in Section 5. The details about the hyperparameters and the training can be found in Appendix A.

### 3.3 Fine-Tuning for Task Adaptation

We experiment with fine-tuning the LMs with an objective similar to the downstream task (WiC). Specifically, we fine-tune the LMs via SPL to classify word occurrence pairs and determine if they share the same supersense. The goal is to investigate how task similarity affects downstream task performance. More precisely, we aim to investigate how adjusting the embedding space to suit the downstream task compares to enhancing it with sense knowledge in terms of performance.

#### 3.3.1 Data Preprocessing

Starting from SemCor as outlined in Section 3.1, we randomly pair sentences containing the same word (either verb or noun) and label the pair depending on whether this word shares the same supersense in both sentences. We only consider words that occur in the corpus with at least two supersenses.

#### 3.3.2 Method

We concatenate the sentence pairs with a separator token in between and feed the result to the LM. We concatenate the embeddings of the two occurrences of the target word (produced by the final layer of the model) and feed the result to a binary classification layer.<sup>9</sup> The LM is fine-tuned via SPL using Cross-Entropy Loss. Details about the hyperparameters and the training can be found in Appendix A.

### 3.4 Isotropization

We apply the isotropization post-processing method proposed by Mu and Viswanath (2018) to

<sup>9</sup>We use as embedding of a word occurrence the average of the embeddings of its subword tokens.

increase the isotropy of the original BERT embeddings. This post-processing method is applied to the embeddings of the test set, which are obtained as described in Section 4.1, to evaluate their topological properties, and also to the WiC dataset to assess task performance with isotropized embeddings. The goal is to evaluate the effects of isotropization in isolation on task performance, without adding additional knowledge.

The method involves three steps. First, the embeddings are centered by computing their mean vector and subtracting the mean vector from each embedding. Then, Principal Component Analysis is applied to the centered embeddings to identify the components that carry high variance. Finally, for some  $k \in \mathbb{N}$  (in our case,  $k = 1$ ), the  $k$  top principal components are removed from each centered embeddings — this is done by subtraction of its projections on these  $k$  directions.<sup>10</sup> Because the principal components are the directions with the highest variance, removing these components helps create a more uniform distribution of directions.

## 4 Evaluation of the Embeddings

We evaluate the embeddings from both BERT and RoBERTa, including the original embeddings (with and without isotropization), embeddings of the models fine-tuned for sense knowledge enhancement using SCL and SPL, and embeddings of the models fine-tuned for task adaptation. The evaluation is based on both the topology of the embedding space and the performance on the WiC task.<sup>11</sup>

### 4.1 Topology of the Embedding Space

We evaluate the quality of the embeddings based on their topological properties using three metrics: *sense-alignment*, *uniformity*, and *isotropy*. This evaluation is performed on the test set (see Section 3.1), using nouns and verbs only. The embeddings are extracted from the final layer of the LM and are L2-normalized before the evaluation. We write  $E = [e_1, \dots, e_N]$  the matrix of all extracted embeddings.

**Sense-Alignment:** This metrics is defined as the average over all supersenses of the average

<sup>10</sup>Our experiments show that removing the first top principal component is enough to substantially increase uniformity and isotropy. For detailed experiments, see Appendix B.

<sup>11</sup>We conducted additional experiments with the Word Sense Disambiguation task (Raganato et al., 2017) and observed performance patterns similar to those seen in the WiC task across different embeddings. Details of these experiments can be found in Appendix C.

|                          | Topological Properties |                 |             | WiC         |                   |
|--------------------------|------------------------|-----------------|-------------|-------------|-------------------|
|                          | Uniformity             | Sense-Alignment | Isotropy    | Threshold   | Classifier        |
| BERT                     | 2.84                   | 0.33            | 0.58        | 63.2        | 55.9 (0.7)        |
| + <i>Isotropization</i>  | <b>3.95</b>            | 0.04            | <b>0.98</b> | <b>63.6</b> | <b>56.7</b> (0.2) |
| + <i>SCL</i>             | 2.81                   | <b>0.72</b>     | <b>0.92</b> | <b>64.7</b> | <b>56.8</b> (0.4) |
| + <i>SPL</i>             | 2.49                   | <b>0.52</b>     | 0.55        | <b>64.0</b> | 55.8 (0.5)        |
| + <i>Task Adaptation</i> | <b>3.37</b>            | 0.22            | <b>0.68</b> | 62.9        | <b>56.5</b> (0.5) |
| RoBERTa                  | 0.73                   | 0.83            | 0.40        | 61.2        | 56.5 (0.5)        |
| + <i>Isotropization</i>  | <b>3.89</b>            | 0.07            | <b>0.97</b> | <b>63.2</b> | <b>56.7</b> (0.5) |
| + <i>SCL</i>             | <b>2.92</b>            | 0.71            | <b>0.90</b> | <b>64.8</b> | <b>59.3</b> (0.3) |
| + <i>SPL</i>             | <b>2.95</b>            | 0.67            | <b>0.70</b> | <b>65.2</b> | <b>58.8</b> (0.6) |
| + <i>Task Adaptation</i> | <b>1.87</b>            | 0.55            | <b>0.46</b> | 61.2        | 56.5 (0.9)        |

Table 1: Evaluation results of different embeddings in terms of topological properties and task performance. Higher values are better for all topological property metrics. Accuracy is reported for *WiC Threshold* and *WiC Classifier*. The mean results over 5 runs are given with standard deviation in brackets for *WiC Classifier*. The values that are better than the original LM embeddings are given in bold. The models with the best *WiC* task performance across various temperature settings are reported (*BERT+SCL*: 0.3 *BERT+SPL*: 0.1, *RoBERTa+SCL*: 0.2 *RoBERTa+SPL*: 4.5).

pairwise cosine similarity for word occurrences with this supersense. Higher values indicate better sense-alignment.

**Uniformity:** This metrics measures the density of the distribution of the embeddings in their vector space (Wang and Isola, 2020). It is defined as

$$-\log \left( \frac{2}{N(N-1)} \sum_{1 \leq i < j \leq N} \exp(-2\|e_i - e_j\|_2^2) \right) \quad (2)$$

Higher values indicate more uniform distributions of the embeddings.

**Isotropy:** This metrics measures the distribution of the embeddings across all directions of the embedding space (Mu and Viswanath, 2018). It is defined as

$$\frac{\min_{u \in U} F(u)}{\max_{u \in U} F(u)} \quad (3)$$

where  $U$  is the set of eigenvectors of  $E^T E$  and  $F$  is the *partition function* introduced by Arora et al. (2016) as

$$F(u) = \sum_{i=1}^N \exp(u \cdot e_i) \quad (4)$$

The isotropy score ranges from 0 to 1 and higher values indicate more isotropic distributions.

## 4.2 WiC Performance

The *WiC* task is a binary classification task where the goal is to determine if two occurrences of a same word form, either both verbs or both nouns, have the same meaning. We evaluate the *WiC* performance of two methods, as proposed in the original paper (Pilehvar and Camacho-Collados, 2019): *WiC Threshold* and *WiC Classifier*. Comparing these two methods allows us to better understand

how different topological properties of the embeddings help to solve this task in two setups, unsupervised and supervised.

For both methods, we extract the embeddings produced by the last 4 layers of the LM and average them to obtain a single embedding per token, as done in previous work (Liu et al., 2021b). Additionally, in cases where the target word is tokenized into subwords by the model tokenizer, we average all subword embeddings to obtain one embedding per word.

**WiC Threshold:** Two occurrences of a word are classified as having the same meaning if the cosine similarity of their embeddings is higher than a fixed threshold. This threshold is tuned on the development set and we report the accuracy achieved with the best threshold on the test data.<sup>12</sup>

**WiC Classifier:** We train a binary classifier to predict whether two occurrences of a word have the same meaning. The classifier takes the concatenation of their embeddings as input. We use a feed-forward network with one hidden layer with ReLU activation.<sup>13</sup> We train five randomly initialized classifiers for each type of embedding and report the mean accuracy.

## 4.3 Results

Table 1 summarizes our results; we report the results for the best temperature for each fine-tuning method, determined by the average performance on both *WiC* task setups.

<sup>12</sup>We have experimented with various similarity metrics, including *dot product* and *Euclidean Distance*, and found that *Cosine Similarity* yields the highest accuracy scores overall.

<sup>13</sup>See Appendix A for details on the hyperparameters and training.

**Language Model-Specific Analysis.** BERT and RoBERTa embeddings exhibit different topological properties. More specifically, RoBERTa embeddings are more sense-aligned, less uniform, and less isotropic. Additionally, embeddings of two LMs differ in performance: BERT embeddings yield better performance on WiC Threshold, whereas RoBERTa embeddings yield better performance on WiC Classifier. Overall, fine-tuned RoBERTa embeddings yield better results on both setups, and the improvements achieved are more significant compared to BERT embeddings.

**Comparison of Fine-Tuning Methods.** For both LMs, isotropization leads to improved performance on both WiC Threshold and WiC Classifier compared to the original embeddings. SCL and SPL produce embeddings that differ in terms of topological properties, however, both methods yield embeddings with similar properties across the two LMs. Compared to SPL, SCL generates embeddings that are more isotropic and more sense-aligned with both models. Uniformity values, however, varies slightly depending on the LM.

Considering WiC performance, SCL achieves the highest average WiC scores with both BERT and RoBERTa, outperforming other methods. With BERT, SPL embeddings outperform the original LM embeddings only on WiC Threshold, showing limited and inconsistent improvements in overall WiC performance. Lastly, fine-tuning for task adaptation does not lead to consistent improvements: it improves performance on WiC Classifier with BERT, but reduces it on WiC Threshold, and shows no gains for either task setup with RoBERTa.

The differences in isotropy and sense-alignment between SCL (higher) and SPL (lower) might be somewhat surprising when considering the two training objectives. SCL works by explicitly pushing closer to each other embeddings of same supersense tokens and pushing farther embeddings of different supersense tokens. SPL, when implemented with a linear classification layer, works in a similar way, although indirectly: each supersense is represented by a vector in the embedding space, and token embeddings are pushed both closer to their supersense vector and farther from other supersense vectors. One obvious difference between the two methods is that we interpret *closer/farther* in terms cosine similarity for SCL and in terms of dot product for SPL. Another one is the use of class vectors in SPL which, depending on their ran-

dom initialization, might cause SPL to affect the embedding space in more arbitrary ways.

### **Topological Properties and Task Performance.**

Our results are in line with previous studies that show the importance of isotropy in downstream task performance (Biś et al., 2021; Gao et al., 2021; Liu et al., 2021a,b; Rajaei and Pilehvar, 2021a). We obtain higher WiC performance with more isotropic embeddings. However, the best performances are achieved with the embeddings that are both more isotropic and sense-aligned with both LMs.

Additionally, high sense-alignment alone does not guarantee strong task performance. RoBERTa embeddings exhibit lower WiC Threshold scores compared to BERT embeddings despite being more sense-aligned. Similarly, fine-tuning BERT with SPL increases sense-alignment without increasing isotropy or uniformity. This only improves performance on WiC Threshold while reducing performance on WiC Classifier. These findings are consistent with previous studies emphasizing the importance of balancing multiple metrics (Wang and Isola, 2020; Wang and Liu, 2021; Gao et al., 2021).

In contrast to these studies, we observe that increased uniformity does not always yield better results and that its effects vary by model. With BERT, the best-performing embeddings have similar (but slightly lower) uniformity values as the original embeddings. With RoBERTa, however, increasing uniformity and isotropy leads to better results. Note that these LMs differ notably in terms of their initial sense-alignment and uniformity values.

Lower uniformity indicates that embeddings are more concentrated in specific regions of the space. In our case, these clusters likely arise from sense-alignment, where the regions are influenced by sense similarities.<sup>14</sup> However, the results show that maintaining the distribution of embeddings (uniformity) as much as possible while increasing sense similarities is more beneficial. This suggests a trade-off between these metrics, and balancing them proves more beneficial. See Appendix D for the visualization of the embedding spaces, illustrating the relation between sense-alignment and uniformity.

<sup>14</sup>The studies mentioned evaluate the impact of uniformity in other contexts. When embeddings are intentionally aligned with specific knowledge, reduced uniformity may arise from meaningful similarities, whereas, e.g., in encoder training, it may arise from unwanted artifacts in the data.

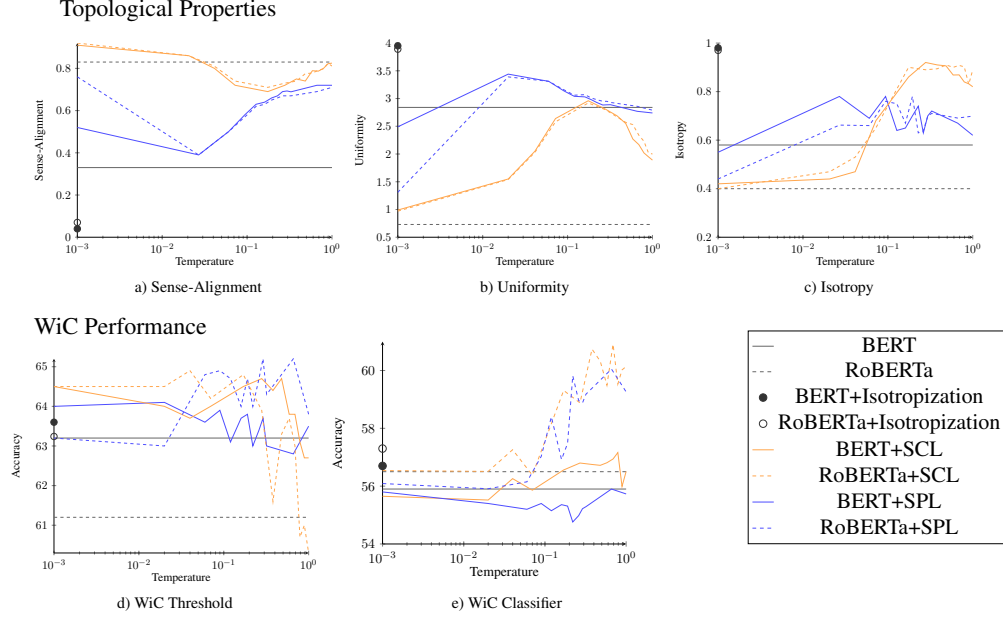


Figure 2: Relation between temperature, different topological properties, and WiC performance. We report the results for the embeddings of LMs fine-tuned with SCL and SPL with different temperature values (normalized), the original LM embeddings, and isotropized LM embeddings (indicated by markers). Three topological properties are studied: *sense-alignment* (a), *uniformity* (b), and *isotropy* (c). For WiC task performance, *WiC Threshold* (d) and *WiC Classifier* (e) performances are studied. The mean results over 5 runs are given for WiC Classifier.

## 5 Relation Between Temperature and Topology of the Embedding Space

In SCL, the temperature parameter of the loss controls the strength of the separation between positive and negative instances. A lower temperature increases the influence of harder-to-separate instances, effectively generating harder negatives. As for SPL, the temperature parameter scales the probability distribution of the classes and higher values of temperature lead to softer probability distributions.

We experiment with different temperature values to investigate the effects of temperature on the topology of the embedding space. Furthermore, we show how the embeddings with different topological properties perform differently on the WiC task. These experiments provide a more in-depth understanding of the fine-tuning methods and gives us a more complete picture of how variations in topological properties relate to task performance.

Temperature is already a parameter in the SCL loss function (see Formula 1). We introduce a temperature parameter into the Cross-Entropy loss by scaling the logits before applying the softmax. In the context of SCL loss, the temperature typically ranges between 0 and 1, whereas in Cross-Entropy loss, the temperature can vary over a wider range,

typically between 0.1 and 100 (Agarwala et al., 2020). We experiment with 13 temperature values for both loss functions ranging between 0.03 and 1 for SCL and between 0.1 and 15 for SPL.<sup>15</sup>

### 5.1 Results

See Figure 2 for the relation between temperature, topological properties, and WiC performance.<sup>16</sup> Overall, temperature affects the topology of the embedding space. A clear trade-off is observed between sense-alignment, isotropy, and uniformity. As sense-alignment decreases, both uniformity and isotropy tend to increase.

**Comparison of Fine-Tuning Methods.** Different fine-tuning methods create embeddings with different properties. Overall, SCL creates more sense-aligned, more isotropic, but less uniform embeddings. On the other hand, SPL creates less sense-aligned, less isotropic, but more uniform embeddings. Furthermore, both methods produce embeddings with similar topological properties with both LMs, despite the initial topological differences between the models.

<sup>15</sup>The selected temperatures for SCL are: 0.03, 0.05, 0.07, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1. The selected temperatures for SPL are: 0.1, 0.5, 1, 1.5, 2, 2.5, 3, 3.5, 4, 4.5, 5, 10, 15.

<sup>16</sup>See Appendix E for complete values.

Considering task performance, both fine-tuning methods lead to performance improvement with RoBERTa, while only SCL leads to performance improvement with BERT. With BERT, SPL embeddings do not outperform the original BERT embeddings at any temperature value on WiC Classifier and they only outperform them in early temperatures on WiC Threshold. Overall, SCL yields better results, demonstrating high performance with both models and task setups.

**Language Model-Specific Analysis.** The performance trends are similar across models, however, the improvements with RoBERTa are more substantial, despite both LMs having embeddings with similar properties after fine-tuning. With RoBERTa, both fine-tuning methods yield more comparable performance, whereas with BERT, SCL shows a clear performance advantage.<sup>17</sup>

**Unsupervised and Supervised Tasks.** Each task exhibits different trends. On WiC Threshold, the fine-tuned embeddings generally outperform the original LM embeddings. Sense-alignment seems to play a significant role after isotropy reaches a certain threshold. Notably, less isotropic but more sense-aligned BERT+SPL embeddings outperform isotropized BERT embeddings. Overall, high performances are achieved when both isotropy and sense-alignment exceed certain levels.

WiC Classifier performance varies more across models and fine-tuned model embeddings do not always surpass the original LM embeddings in terms of performance. Both high levels of isotropy and sense-alignment seem important. For example, with BERT, surpassing the isotropized embeddings is challenging and only happens at the latest temperatures with SCL, when isotropy and sense-alignment are both significantly high. Similarly, highly sense-aligned original RoBERTa embeddings already start with better performance than the BERT embeddings and the best performances are achieved with RoBERTa when both properties are significantly high.

## 6 Conclusion

In this study, we explore the relation between different fine-tuning methods, the topology of the resulting embedding spaces and the downstream task performance in the context of sense knowledge

enhancement with two pre-trained language models (LMs), BERT and RoBERTa. We focus on three topological properties: *sense-alignment*, *uniformity*, and *isotropy*.

Our comparison of two fine-tuning methods, *Supervised Contrastive Learning (SCL)* and *Supervised Predictive Learning (SPL)*, shows that these methods create embedding spaces that differ in terms of topological properties and the resulting embeddings achieve different task performance. Although the embeddings of BERT and RoBERTa differ in their initial properties, fine-tuning methods affect the embedding spaces of both LMs in similar ways, resulting in embeddings with similar properties. Overall, SPL — the standard fine-tuning method — shows mixed results, with effectiveness varying by LM. It produces embeddings that outperform the original LM embeddings in most cases, though not consistently. In contrast, SCL demonstrates more consistent improvements with both LMs and generally outperforms SPL.

With respect to topological properties, our results show that the embeddings with increased isotropy achieve better task performance. However, the best-performing embeddings are those with high sense-alignment and isotropy. Embeddings with only high sense-alignment do not always yield better results and can underperform compared to the original LM embeddings. Furthermore, we observe a trade-off between uniformity and sense-alignment, with optimal results achieved by balancing the two. Overall, our results highlight the importance of balancing all three properties for optimal task performance.

Moreover, our results show that the degree to which these topological properties are beneficial in a downstream task differs between a supervised and an unsupervised system. Both sense-alignment and isotropy contribute to performance, but their relative importance varies across task setups. Finally, our findings suggest that fine-tuning an LM on a task similar to the downstream task does not necessarily produce embeddings that improve downstream task performance.

In conclusion, our findings highlight the importance of topological properties of the embeddings during knowledge enhancement, demonstrating that fine-tuning an LM without considering these properties may not always lead to improved downstream task performance.

<sup>17</sup>The difference in improvement between the LMs may be influenced by factors beyond the scope of our current analysis, warranting further investigation.

## Limitations

While the results of this study offer valuable insights, several limitations must be acknowledged. The main limitation of this study is that our experiments only focus on English, which limits the generalization of our findings to other languages. However, we expect similar results with other languages as topological properties of the embeddings are more closely linked to factors such as learning objectives, loss functions used for the training of the models.

Similarly, our experiments rely on WordNet supersedes and the SemCor corpus, which limits the generalization of our results to other sense inventories or datasets. However, these resources are widely used and highly valuable for word sense tasks.

We further acknowledge that there may be other properties influencing the success of language model embeddings, and our study focuses on only three specific properties. However, isotropy and uniformity are commonly used metrics, especially in terms of expressiveness. Furthermore, our study offers potential explanations for the observed performance differences across different types of embeddings with these properties.

Finally, we rely on Cosine Similarity as a similarity metric for both training and evaluation of the models. However, this metric is the most widely used and successful semantic similarity metric. Using this metric in both stages does not influence the core findings of our study; instead, it highlights the importance of isotropy.

## Acknowledgement

This study is funded by the DFG project “Coercion and Copredication as Flexible Frame Composition”. We would like to thank the anonymous reviewers for their valuable comments.

## References

- Atish Agarwala, Jeffrey Pennington, Yann Dauphin, and Sam Schoenholz. 2020. [Temperature check: theory and practice for training models with softmax-cross-entropy losses](#). *arXiv preprint arXiv:2010.07344*.
- Carlos Santos Armendariz, Matthew Purver, Senja Polak, Nikola Ljubešić, Matej Ulčar, Ivan Vulić, and Mohammad Taher Pilehvar. 2020. [SemEval-2020 task 3: Graded word similarity in context](#). In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 36–49, Barcelona (online). International Committee for Computational Linguistics.
- Sanjeev Arora, Yuanzhi Li, Yingyu Liang, Tengyu Ma, and Andrej Risteski. 2016. [A latent variable model approach to pmi-based word embeddings](#). *Transactions of the Association for Computational Linguistics*, 4:385–399.
- Geetanjali Bihani and Julia Rayz. 2021. [Low anisotropy sense retrofitting \(LAsER\) : Towards isotropic and sense enriched representations](#). In *Proceedings of Deep Learning Inside Out (DeeLIO): The 2nd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, pages 81–95, Online. Association for Computational Linguistics.
- Steven Bird and Edward Loper. 2004. [NLTK: The natural language toolkit](#). In *Proceedings of the ACL Interactive Poster and Demonstration Sessions*, pages 214–217, Barcelona, Spain. Association for Computational Linguistics.
- Daniel Biś, Maksim Podkorytov, and Xiuwen Liu. 2021. [Too much in common: Shifting of embeddings in transformer language models and its implications](#). In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5117–5130, Online. Association for Computational Linguistics.
- Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. [A simple framework for contrastive learning of visual representations](#). In *International conference on machine learning*, pages 1597–1607. PMLR.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [BERT: Pre-training of deep bidirectional transformers for language understanding](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
- Kawin Ethayarajh. 2019. [How contextual are contextualized word representations? Comparing the geometry of BERT, ELMo, and GPT-2 embeddings](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 55–65, Hong Kong, China. Association for Computational Linguistics.
- Manaal Faruqui, Jesse Dodge, Sujay Kumar Jauhar, Chris Dyer, Eduard Hovy, and Noah A. Smith. 2015. [Retrofitting word vectors to semantic lexicons](#). In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1606–1615, Denver, Colorado. Association for Computational Linguistics.

- Christiane Fellbaum. 1998. *WordNet: An electronic lexical database*. MIT press.
- Jun Gao, Di He, Xu Tan, Tao Qin, Liwei Wang, and Tie-Yan Liu. 2019. Representation degeneration problem in training natural language generation models. In *International Conference on Learning Representations*.
- Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Aina Garí Soler and Marianna Apidianaki. 2020. MULLETSEM at SemEval-2020 task 3: Fine-tuning BERT for lexical meaning. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 158–165, Barcelona (online). International Committee for Computational Linguistics.
- Beliz Gunel, Jingfei Du, Alexis Conneau, and Ves Stoyanov. 2020. Supervised contrastive learning for pre-trained language model fine-tuning. In *International conference on machine learning*.
- Linmei Hu, Zeyi Liu, Ziwang Zhao, Lei Hou, Liqiang Nie, and Juanzi Li. 2023. A survey of knowledge enhanced pre-trained language models. *IEEE Transactions on Knowledge and Data Engineering*.
- Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschiot, Ce Liu, and Dilip Krishnan. 2020. Supervised contrastive learning. *Advances in neural information processing systems*, 33:18661–18673.
- Anne Lauscher, Ivan Vulić, Edoardo Maria Ponti, Anna Korhonen, and Goran Glavaš. 2020. Specializing unsupervised pretraining models for word-level semantic similarity. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1371–1383, Barcelona, Spain (Online). International Committee on Computational Linguistics.
- Yoav Levine, Barak Lenz, Or Dagan, Ori Ram, Dan Padnos, Or Sharir, Shai Shalev-Shwartz, Amnon Shashua, and Yoav Shoham. 2020. SenseBERT: Driving some sense into BERT. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4656–4667, Online. Association for Computational Linguistics.
- Fangyu Liu, Ivan Vulić, Anna Korhonen, and Nigel Collier. 2021a. Fast, effective, and self-supervised: Transforming masked language models into universal lexical and sentence encoders. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1442–1459, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Qianchu Liu, Fangyu Liu, Nigel Collier, Anna Korhonen, and Ivan Vulić. 2021b. MirrorWiC: On eliciting word-in-context representations from pretrained language models. In *Proceedings of the 25th Conference on Computational Natural Language Learning*, pages 562–574, Online. Association for Computational Linguistics.
- Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.
- George A. Miller, Claudia Leacock, Randee Teng, and Ross T. Bunker. 1993. A semantic concordance. In *Human Language Technology: Proceedings of a Workshop Held at Plainsboro, New Jersey, March 21-24, 1993*.
- Anna Mosolova, Marie Candito, and Carlos Ramisch. 2024. Injecting Wiktionary to improve token-level contextual representations using contrastive learning. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 34–41, St. Julian's, Malta. Association for Computational Linguistics.
- Jiaqi Mu and Pramod Viswanath. 2018. All-but-the-top: Simple and effective postprocessing for word representations. In *International Conference on Learning Representations*.
- Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32.
- Matthew E. Peters, Mark Neumann, Robert Logan, Roy Schwartz, Vidur Joshi, Sameer Singh, and Noah A. Smith. 2019. Knowledge enhanced contextual word representations. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 43–54, Hong Kong, China. Association for Computational Linguistics.
- Mohammad Taher Pilehvar and Jose Camacho-Collados. 2019. WiC: the word-in-context dataset for evaluating context-sensitive meaning representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1267–1273, Minneapolis, Minnesota. Association for Computational Linguistics.
- Alessandro Raganato, Jose Camacho-Collados, and Roberto Navigli. 2017. Word sense disambiguation: A unified evaluation framework and empirical comparison. In *Proceedings of the 15th Conference of*

- the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 99–110, Valencia, Spain. Association for Computational Linguistics.
- Sara Rajaei and Mohammad Taher Pilehvar. 2021a. [A cluster-based approach for improving isotropy in contextual embedding space](#). In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 575–584, Online. Association for Computational Linguistics.
- Sara Rajaei and Mohammad Taher Pilehvar. 2021b. [How does fine-tuning affect the geometry of embedding space: A case study on isotropy](#). In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 3042–3049, Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Younes Samih and Laura Kallmeyer. 2023. [Unsupervised semantic frame induction revisited](#). In *Proceedings of the 15th International Conference on Computational Semantics*, pages 89–93, Nancy, France. Association for Computational Linguistics.
- Dilin Wang, Chengyue Gong, and Qiang Liu. 2019a. [Improving neural language modeling via adversarial training](#). In *International Conference on Machine Learning*, pages 6555–6565. PMLR.
- Feng Wang and Huaping Liu. 2021. [Understanding the Behaviour of Contrastive Loss](#). In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2495–2504, Los Alamitos, CA, USA. IEEE Computer Society.
- Lingxiao Wang, Jing Huang, Kevin Huang, Ziniu Hu, Guangtao Wang, and Quanquan Gu. 2019b. [Improving neural language generation with spectrum control](#). In *International Conference on Learning Representations*.
- Tongzhou Wang and Phillip Isola. 2020. [Understanding contrastive representation learning through alignment and uniformity on the hypersphere](#). In *International conference on machine learning*, pages 9929–9939. PMLR.
- Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. [Transformers: State-of-the-art natural language processing](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.
- Yuanmeng Yan, Rumei Li, Sirui Wang, Fuzheng Zhang, Wei Wu, and Weiran Xu. 2021. [ConSERT: A contrastive framework for self-supervised sentence representation transfer](#). In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5065–5075, Online. Association for Computational Linguistics.
- Deniz Ekin Yavas. 2024. [Assessing the significance of encoded information in contextualized representations to word sense disambiguation](#). In *Proceedings of the Third Workshop on Understanding Implicit and Underspecified Language*, pages 42–53, Malta. Association for Computational Linguistics.
- Deniz Ekin Yavas, Timothée Bernard, Laura Kallmeyer, and Benoît Crabbé. 2024. [Improving word sense induction through adversarial forgetting of morphosyntactic information](#). In *Proceedings of the 13th Joint Conference on Lexical and Computational Semantics (\*SEM 2024)*, pages 238–251, Mexico City, Mexico. Association for Computational Linguistics.
- David Yenicelek, Florian Schmidt, and Yannic Kilcher. 2020. [How does BERT capture semantics? a closer look at polysemous words](#). In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 156–162, Online. Association for Computational Linguistics.
- Haode Zhang, Haowen Liang, Yuwei Zhang, Li-Ming Zhan, Xiao-Ming Wu, Xiaolei Lu, and Albert Lam. 2022. [Fine-tuning pre-trained language models for few-shot intent detection: Supervised pre-training and isotropization](#). In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 532–542, Seattle, United States. Association for Computational Linguistics.
- Zhong Zhang, Chongming Gao, Cong Xu, Rui Miao, Qinli Yang, and Junming Shao. 2020. [Revisiting representation degeneration problem in language modeling](#). In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 518–527, Online. Association for Computational Linguistics.

## A Model Hyperparameters and Training Details

The training of the models is done on NVIDIA RTX A5000 GPU, with a single GPU. The PyTorch library (Paszke et al., 2019) is utilized for the training. We use BERT (*base-uncased*) for English with parameter size 110M and RoBERTa (*base*) for English with parameter size 125M, which are accessed via the Transformers library (Wolf et al., 2020). The SemCor corpus, which is the data used for fine-tuning, is accessed via the NLTK Library (Bird and Loper, 2004). Finally, the WiC

Dataset, evaluation data, is accessed via <https://super.gluebenchmark.com/tasks>. Both SemCor and WiC are created for the English language.

Licenses for the model and data used:

- WiC: Creative Commons Attribution-NonCommercial 4.0 License.
- BERT: Apache license 2.0

## A.1 Fine-Tuning for Sense Knowledge Enhancement

### A.1.1 Supervised Contrastive Learning

- **Optimizer:** Adam
- **Batch Size:** 8
- **Learning Rate:**
  - BERT:  $1e - 05$ ,
  - RoBERTa:  $1e - 05$ ,  $1e - 04$  after  $\tau=0.5$ ,
- **Number of Epochs (best  $\tau$ ):**
  - BERT: 4,
  - RoBERTa: 4,
- **$\tau$  (best):**
  - BERT: 0.3,
  - RoBERTa: 0.2,
- **Loss:** SCL
- We employ early stopping based on validation loss.

### A.1.2 Supervised Predictive Learning

- **Classification Head:** A dense layer with a Softmax activation function
- **Optimizer:** Adam
- **Batch Size:** 8
- **Learning Rate:**
  - BERT:  $1e - 05$ ,
  - RoBERTa:  $1e - 05$ ,  $1e - 04$  after  $\tau=4$ ,
- **Number of Epochs (best  $\tau$ ):**
  - BERT: 3,
  - RoBERTa: 2,
- **$\tau$  (best):**
  - BERT: 0.1,
  - RoBERTa: 4.5,
- **Loss:** Cross-Entropy
- We employ early stopping based on validation loss.

## A.2 Fine-Tuning for Task Adaptation

- **Classification Head:** A dense layer with a Sigmoid activation function
- **Optimizer:** Adam
- **Batch Size:** 8

- **Learning Rate:**

- BERT:  $5e - 06$ ,
- RoBERTa:  $1e - 06$

- **Number of Epochs:**

- BERT: 2,
- RoBERTa: 4,

- **Loss:** Cross-Entropy

- We employ early stopping based on validation loss.

### A.3 WiC Task

- A feed-forward network with one hidden layer with ReLU activation and an output layer with Sigmoid activation.
- **Optimizer:** Adam
- **Batch Size:** 32
- **Learning Rate:**  $1e - 04$
- **Drop-out:** 0.2
- We employ early stopping based on validation loss.
- We train a classifier five times using each type of embedding and report the mean test accuracy.

## B Isotropization Experiments

For the isotropization post-processing method, we test the effect of removing various numbers of top components from the BERT embeddings, specifically 1, 5, 10, 15, and 20 components. The quality of the resulting embeddings is presented in Table 2.

This method successfully increases the isotropy and uniformity of BERT embeddings. The number of top components removed influences the trade-off between isotropy and the retention of semantic information in the embeddings. Isotropy and uniformity improve after removing just the first top component, with both continuing to improve as more components are removed. Furthermore, we notice that sense-alignment drops significantly only after the first top component is removed. As more components are removed, less semantic information is preserved, and sense-alignment continues to reduce.

## C Word Sense Disambiguation Task

The Word Sense Disambiguation (WSD) task (Raganato et al., 2017) involves selecting the correct WordNet sense label for a target word in context from its candidate senses. Similar to our approach,

| Model Type          | #  | Uniformity | Sense-Alignment | Isotropy |
|---------------------|----|------------|-----------------|----------|
| BERT                | -  | 2.840      | 0.332           | 0.588    |
| BERT+Isotropization |    |            |                 |          |
|                     | 1  | 3.958      | 0.045           | 0.986    |
|                     | 5  | 3.967      | 0.031           | 0.988    |
|                     | 10 | 3.971      | 0.022           | 0.993    |
|                     | 15 | 3.972      | 0.018           | 0.995    |
|                     | 20 | 3.973      | 0.015           | 0.995    |

Table 2: Topological properties of the BERT embeddings and isotropized BERT embeddings with different numbers of components removed. We report 3 topological properties: *sense-alignment*, *uniformity*, and *isotropy*. Higher values are better for all topological properties.



Figure 3: The t-SNE visualizations of the test data using embeddings from different models. Models: BERT embeddings (left), BERT+SCL embeddings with  $\tau = 0.2$  (middle), BERT+SCL embeddings with  $\tau = 1.0$  (right). Different colors refer to different supersenses.

| Model            | WSD         |
|------------------|-------------|
| BERT             | 48.1        |
| +Isotropization  | 47.9        |
| +SCL             | <b>48.9</b> |
| +SPL             | <b>48.8</b> |
| +Task Adaptation | <b>48.7</b> |
| RoBERTa          | 49.0        |
| +Isotropization  | <b>50.4</b> |
| +SCL             | <b>53.3</b> |
| +SPL             | <b>52.2</b> |
| +Task Adaptation | <b>50.6</b> |

Table 3: WSD results using different embeddings. *F1* is reported. The values that are better than the original LM embeddings are given in bold. The results of the models with the best WiC task performance across various temperature settings are reported (*BERT+SCL*: 0.3 *BERT+SPL*: 0.1, *RoBERTa+SCL*: 0.2 *RoBERTa+SPL*: 4.5).

SemCor is used as the training corpus in the original task. Following previous work (Liu et al., 2021b), we design a one-shot setting in which each sense is represented by a single example from

WordNet. We compute the cosine similarity between the embedding of the target word in the example sentences and that of the test instance, and select the sense label with the highest similarity. We exclude the senses that do not have any examples in WordNet.

The results with the embeddings of the LMs and fine-tuned LMs (best temperature) can be seen in Table 3. While isotropization does not always improve the performance, the SCL fine-tuning method achieves the best results with both LMs.

## D Visualization of the Embedding Spaces

Since uniformity is measured using the pairwise similarities between embeddings, the decrease in uniformity observed in sense-enhanced embeddings may be associated with the formation of dense sense clusters. As sense-alignment increases and uniformity decreases, instances of the same sense may become closer. This relation can be captured with t-SNE visualization, as it is useful for capturing the local relations in the data.

The t-SNE visualizations of the test set using embeddings from different models are shown in

Figure 3. The models and the topological properties of their embeddings are as follows:

- **BERT:**  
Sense-Alignment: 0.33, Uniformity: 2.84
- **BERT+SCL with  $\tau = 0.2$ :**  
Sense-Alignment: 0.72, Uniformity: 2.96
- **BERT+SCL with  $\tau = 1.0$ :**  
Sense-Alignment: 0.82, Uniformity: 1.89

We observe that the embeddings of BERT+SCL ( $\tau = 1.0$ ) with reduced uniformity show to form more dense clusters of same sense instances.

## E Temperature Details

The complete topological property and WiC performance values for embeddings extracted from models with different temperatures are provided in Table 4.

| Model Type  | Temperature | Topological Properties |                 |          | WiC Performance |                    |
|-------------|-------------|------------------------|-----------------|----------|-----------------|--------------------|
|             |             | Uniformity             | Sense-Alignment | Isotropy | Threshold       | Classifier         |
| BERT        | -           | -2.83                  | 0.33            | 0.58     | 63.2            | 55.90 (0.7)        |
| BERT+SPL    |             |                        |                 |          |                 |                    |
|             | 0.1         | 2.49                   | 0.52            | 0.55     | 64.0            | 55.8 (0.5)         |
|             | 0.5         | 3.44                   | 0.39            | 0.78     | 64.1            | 55.4 (0.3)         |
|             | 1           | 3.31                   | 0.50            | 0.69     | 63.6            | 55.28 (0.8)        |
|             | 1.5         | 3.14                   | 0.58            | 0.78     | 63.9            | 55.43 (0.5)        |
|             | 2           | 3.04                   | 0.63            | 0.64     | 63.1            | 55.15 (0.1)        |
|             | 2.5         | 3.03                   | 0.64            | 0.65     | 63.7            | 55.36 (0.8)        |
|             | 3           | 2.98                   | 0.66            | 0.70     | 63.8            | 55.31 (0.2)        |
|             | 3.5         | 2.93                   | 0.67            | 0.74     | 63.0            | 54.76 (0.4)        |
|             | 4           | 2.88                   | 0.69            | 0.63     | 63.4            | 54.98 (0.4)        |
|             | 4.5         | 2.88                   | 0.69            | 0.70     | 63.7            | 55.22 (0.7)        |
|             | 5           | 2.88                   | 0.68            | 0.72     | 63.0            | 55.29 (0.6)        |
|             | 10          | 2.77                   | 0.72            | 0.67     | 62.8            | 55.9 (0.8)         |
|             | 15          | 2.74                   | 0.72            | 0.62     | 63.5            | 55.73 (0.4)        |
| BERT+SCL    |             |                        |                 |          |                 |                    |
|             | 0.03        | 0.99                   | 0.91            | 0.42     | 64.5            | 55.65 (0.3)        |
|             | 0.05        | 1.55                   | 0.86            | 0.44     | 64.0            | 55.52 (0.6)        |
|             | 0.07        | 2.05                   | 0.80            | 0.47     | 63.7            | 56.26 (0.4)        |
|             | 0.1         | 2.64                   | 0.72            | 0.68     | 64.0            | 55.86 (1.1)        |
|             | 0.2         | 2.96                   | 0.69            | 0.86     | 64.5            | 56.56 (0.5)        |
|             | 0.3         | 2.81                   | 0.72            | 0.92     | 64.7            | 56.8 (0.4)         |
|             | 0.4         | 2.69                   | 0.75            | 0.91     | 64.4            | 56.75 (0.4)        |
|             | 0.5         | 2.55                   | 0.74            | 0.90     | 64.7            | 56.72 (0.3)        |
|             | 0.6         | 2.27                   | 0.78            | 0.86     | 63.8            | 56.81 (0.6)        |
|             | 0.7         | 2.17                   | 0.78            | 0.86     | 63.8            | 56.94 (0.7)        |
|             | 0.8         | 2.02                   | 0.80            | 0.84     | 63.1            | 57.16 (0.7)        |
|             | 0.9         | 1.95                   | 0.82            | 0.83     | 62.7            | 55.99 (0.4)        |
|             | 1           | 1.89                   | 0.82            | 0.82     | 62.7            | 56.51 (0.3)        |
| RoBERTa     | -           | 0.73                   | 0.83            | 0.40     | 61.2            | 56.50 (0.5)        |
| RoBERTa+SPL |             |                        |                 |          |                 |                    |
|             | 0.1         | 1.31                   | 0.76            | 0.44     | 63.2            | 56.09 (0.4)        |
|             | 0.5         | 3.39                   | 0.39            | 0.66     | 63.0            | 55.91 (0.6)        |
|             | 1           | 3.31                   | 0.50            | 0.66     | 64.8            | 56.15 (0.5)        |
|             | 1.5         | 3.16                   | 0.57            | 0.76     | 64.9            | 57.03 (0.4)        |
|             | 2           | 3.06                   | 0.62            | 0.75     | 64.7            | 58.40 (0.6)        |
|             | 2.5         | 3.07                   | 0.63            | 0.67     | 64.0            | 56.92 (0.5)        |
|             | 3           | 3.01                   | 0.65            | 0.78     | 64.7            | 57.48 (0.9)        |
|             | 3.5         | 2.98                   | 0.66            | 0.63     | 64.0            | 59.82 (0.2)        |
|             | 4           | 2.95                   | 0.67            | 0.63     | 64.7            | 58.96 (0.8)        |
|             | 4.5         | 2.95                   | 0.67            | 0.70     | <b>65.2</b>     | 58.88 (0.6)        |
|             | 5           | 2.93                   | 0.67            | 0.71     | 64.3            | 59.18 (0.6)        |
|             | 10          | 2.86                   | 0.69            | 0.69     | <b>65.2</b>     | <b>60.05</b> (1.3) |
|             | 15          | 2.79                   | 0.71            | 0.69     | 63.8            | 59.26 (0.8)        |
| RoBERTa+SCL |             |                        |                 |          |                 |                    |
|             | 0.03        | 0.97                   | 0.92            | 0.40     | 64.5            | 56.54 (0.3)        |
|             | 0.05        | 1.54                   | 0.86            | 0.47     | 64.5            | 56.51 (0.3)        |
|             | 0.07        | 2.03                   | 0.81            | 0.53     | <b>64.9</b>     | 57.26 (0.4)        |
|             | 0.1         | 2.58                   | 0.74            | 0.65     | 64.2            | 56.38 (0.3)        |
|             | 0.2         | 2.92                   | 0.71            | 0.90     | 64.8            | 59.31 (0.3)        |
|             | 0.3         | 2.80                   | 0.73            | 0.89     | 63.9            | 58.88 (0.6)        |
|             | 0.4         | 2.67                   | 0.74            | 0.89     | 61.6            | 60.74 (0.3)        |
|             | 0.5         | 2.57                   | 0.77            | 0.91     | 63.3            | 60.32 (0.1)        |
|             | 0.6         | 2.53                   | 0.77            | 0.90     | 63.7            | 59.49 (0.2)        |
|             | 0.7         | 2.34                   | 0.79            | 0.90     | 63.0            | <b>60.91</b> (0.3) |
|             | 0.8         | 2.23                   | 0.79            | 0.90     | 60.7            | 59.53 (0.3)        |
|             | 0.9         | 2.02                   | 0.82            | 0.83     | 61.0            | 60.02 (0.7)        |
|             | 1           | 2.00                   | 0.81            | 0.89     | 60.3            | 60.14 (0.4)        |

Table 4: Topological properties and WiC results of the embeddings of different models with different temperature parameter values. The results for original LM embeddings (BERT and RoBERTa) are given for comparison. Accuracy is given for *WiC Threshold* and *WiC Classifier* results. For WiC Classifier, the mean results over 5 runs are given with standard deviation in brackets. Higher values are better for all topological property metrics.



# Chapter 7

## Discussion and Conclusion

This chapter summarizes and discusses the findings of the five research papers (Section 7.1) and discusses the contributions (Section 7.2), limitations (Section 7.3), and directions for future research (Section 7.4). It concludes with broader implications of the findings (Section 7.5).

### 7.1 Summary and Discussion of the Findings

This section presents a detailed summary of the findings for each research question investigated in this thesis, addressed across five research papers in Chapters 2-6. Section 7.1.1 presents the findings for RQ1, Section 7.1.2 for RQ2, Section 7.1.3 for RQ3, and Section 7.1.4 for RQ4. Finally, the section concludes with a synthesis that connects these findings in Section 7.1.5.

The research questions and hypotheses addressed in this thesis are as follows:

- **RQ1:** Do contextualized word embeddings encode sufficient information for detecting copredication? How difficult is copredication identification for expert human annotators?
  - **H1a:** Contextualized word embeddings encode sufficient information for copredication detection and can serve as effective input features to classifiers.
  - **H1b:** If contextualized word embeddings encode sufficient information for copredication detection (H1a), the embeddings of multilingual Pre-trained Transformer Language Models (PLMs) should enable zero-shot transfer across languages.
  - **H1c:** Because both meaning facets are always available with inherently polysemous nouns, distinguishing contexts where only one facet is active from contexts where both are simultaneously active (copredication) is challenging even for expert human annotators, as reflected in inter-annotator agreement scores.
- **RQ2:** Which contextual features determine embedding similarity between word instances, and are they sense-relevant? Do sense-irrelevant features affect embedding similarity differently across homonymous and polysemous senses?

- **H2a:** The features determining embedding similarity between word instances are not consistently sense-relevant.
- **H2b:** Sense-irrelevant features primarily determine embedding similarity in the case of polysemy due to high contextual overlap. In contrast, sense-relevant features determine embedding similarity in the case of homonymy due to low contextual overlap.
- **RQ3:** How can we successfully remove sense-irrelevant features from contextualized word embeddings of PLMs? What impact does this removal have on unsupervised lexical ambiguity resolution performance?
  - **H3a:** Sense-irrelevant features can be successfully removed from contextualized word embeddings, while preserving their overall embedding quality.
  - **H3b:** Although some sense-irrelevant features may correlate with senses for some words, overall they introduce noise. Therefore, removing these features will improve performance despite exceptions at the individual word level.
- **RQ4:** Can successful sense-enhanced embeddings be created through fine-tuning? How do topological properties of the embeddings (sense-alignment, isotropy, uniformity) relate to task performance? How do Supervised Contrastive Learning (SCL) and Supervised Predictive Learning (SPL) differ in their effects on the topology of the PLM embedding space?
  - **H4a:** Successful sense-enhanced embeddings can be created through fine-tuning PLMs with both SPL and SCL.
  - **H4b:** Improving isotropy and uniformity alongside sense-alignment leads to better performance in lexical ambiguity resolution.
  - **H4c:** SCL and SPL differ in their effect on the topology of the embedding space of PLMs, with SCL expected to achieve a better balance among sense-alignment, isotropy, and uniformity for performance.

### 7.1.1 Copredication as Complex Ambiguity Phenomenon (RQ1)

Two research papers address RQ1: *Cross-lingual Copredication Detection* (Yavas et al., 2023) and *Copredication Corpus Annotation* (Chen et al., 2025b). RQ1 asks whether contextualized word embeddings encode sufficient information for detecting copredication with inherently polysemous nouns. Additionally, to contextualize the computational challenge, RQ1 asks how difficult copredication identification is for expert human annotators.

To address the first part of RQ1, the research paper *Cross-lingual Copredication Detection* develops and evaluates an automatic copredication detection method. This method utilizes classifiers that use contextualized word embeddings as inputs. For this purpose, individual classifiers are trained to automatically classify the semantic argument types of predicates (verbs and adjectives) in context for five semantic types: INFORMATION, EVENT, ARTIFACT, HUMAN, and LOCATION. These

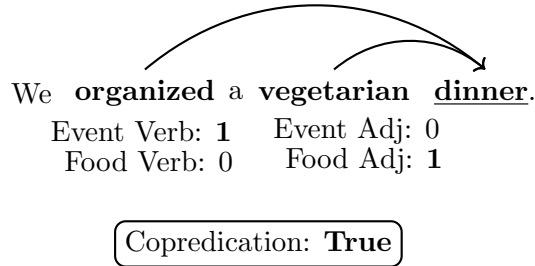


Figure 7.1: Copredication detection method. Given a sentence where both a verb and an adjective target the same noun, the method uses classifiers to detect whether these predicates select different semantic types.

classifiers use contextualized word embeddings of both the predicate and its argument in a sentence as input to decide whether the predication selects a specific semantic type.

These classifiers are then used for copredication detection with inherently polysemous nouns. The paper focuses specifically on *verb+adjective* copredication constructions with FOOD•EVENT nouns (e.g., ‘dinner’, ‘lunch’, ‘meal’). Example (1) illustrates this with ‘dinner’, a FOOD•EVENT noun whose FOOD facet is selected by ‘vegetarian’ and whose EVENT facet is selected by ‘organize’. The copredication detection method works as follows: individual classifiers are applied to the predicates that target the same noun in a sentence, and copredication is detected when predicates are classified as selecting different semantic types, as illustrated in Figure 7.1.

(1) We *organized* a *vegetarian* **dinner**.

The paper tests the performance of individual classifiers and copredication detection on Italian using Italian BERT embeddings as input. Furthermore, it tests whether this information transfers across languages in a zero-shot setting by training multilingual classifiers using the embeddings of multilingual models mBERT and XLM-RoBERTa. These classifiers are trained on Italian data, then applied in a zero-shot setting to English, German, Turkish, and Chinese for evaluation. The multilingual classifiers are then used for copredication detection for all languages.

The results show that all semantic type classifiers perform well for both predicate types across monolingual and cross-lingual settings. Similarly, copredication detection results are promising for all languages (average geometric mean of specificity and sensitivity = 0.49, random baseline = 0.25). These findings suggest both that contextualized word embeddings encode sufficient information for detecting this complex phenomenon and that this information transfers successfully across languages in zero-shot settings. These results confirm both H1a and H1b.

Notably, the results reveal a challenge in detecting the absence of copredication. Even when both predicates select the same facet, i.e., in negative cases of copredication, copredication is sometimes incorrectly detected. Moreover, this tendency occurs only with inherently polysemous nouns, not with simple type nouns. In (2), both predicates in the sentences (‘eat’ and ‘cold’) select the FOOD facet, therefore neither sentence is a copredication. However, when there is an inherently polysemous noun involved in the predications (‘lunch’) as in (2-a) as opposed to a single type noun (‘soup’) as in (2-b), copredication is detected.

- (2) a. **lunch** (FOOD•EVENT)  
       It’s depressing to *eat* a *cold* **lunch**.                      → Copredication Detected  
       b. **soup** (FOOD):  
       It’s depressing to *eat* a *cold* **soup**.                      → Copredication Not Detected

This tendency is slightly overcome by adding additional training data with FOOD•EVENT nouns to the training data of individual classifiers, since the original training data lacks examples with inherently polysemous nouns. This improves cross-lingual copredication detection performance (average geometric mean = 0.57 compared to 0.49). However, the difficulty in detecting the absence of copredication confirms that both facets are potentially available with inherently polysemous nouns regardless of the copredication context and that their ambiguous nature is captured by the embeddings, which contributes to the complexity of the task.

**Human Performance on Copredication Annotation.** The research paper *Copredication Corpus Annotation* details the construction and annotation of CoCoCorpus<sup>1</sup>, a copredication corpus for English. The paper provides a detailed analysis of the annotation, which sheds light on the human performance aspect of RQ1. The paper focuses on five nouns for each of three types of inherent polysemy: FOOD•EVENT, INFORMATION•EVENT, and PHYSICAL\_OBJECT•INFORMATION. Similarly to the *Cross-Lingual Copredication Detection* paper, it focuses on *verb+adjective* copredication construction.

Copredication annotation is conducted in four steps. First, target constructions containing the selected nouns are automatically extracted from the BookCorpus. Specifically, these are the constructions where the target nouns are predicated by both a verb and an adjective. These candidates are then filtered through preliminary annotation using the classifiers from the paper *Cross-Lingual Copredication Detection*. In the third step, two expert annotators manually annotate the filtered constructions. Finally, disagreements are resolved through discussion, leading to gold standard labels.

The annotation yields low to moderate inter-annotator agreement across different types of inherent polysemy (average Cohen’s Kappa = 0.58). Furthermore, the paper provides a detailed analysis of annotator disagreements. This reveals that disagreements stem from several factors, including unclear distinctions between the facets of inherently polysemous nouns, suggesting that determining the active facet is challenging even for human experts, confirming H1c. Overall, these findings demonstrate that copredication identification poses challenges not only for computational models but also for expert human annotators, highlighting the inherently complex nature of the phenomenon. This contextualizes the performance of computational models, as we cannot expect computational models to perform substantially better than human experts on such an inherently difficult task.

### 7.1.2 Contextual Features and Sense-Relevance (RQ2)

RQ2 is addressed in *Feature Relevance for WSD* (Yavas, 2024). RQ2 asks which contextual features determine embedding similarity between word instances, whether these features are sense-relevant, and whether the sense-irrelevant features affect

---

<sup>1</sup>Available at: <https://github.com/CoCoCo-Project>

embedding similarity differently across homonymy and polysemy due to their different levels of contextual overlap. To address these questions, the paper *Feature Relevance for WSD* conducts a cluster explainability experiment to systematically identify the sentence context features that lead to cluster formation in unsupervised Word Sense Disambiguation (WSD) and assess the relevance of these features to distinguishing senses.

The cluster explainability method works in five steps:

1. The target word instances are first clustered using their contextualized word embeddings from BERT.
2. These sentences are then represented by their *sentence context features*: the bag-of-words representation of the sentence, morphological properties, syntactic role, and position of the target word, and the part-of-speech tags of the target word’s neighbors.
3. A label is assigned to each sentence based on the cluster it belongs to.
4. *Classifiers for cluster assignment* are trained to recreate these clusters using sentence context features as input and cluster labels as targets.
5. To identify the features that drive cluster formation, *recursive feature elimination* is applied to these classifiers, and the top 10 features per word are selected.
6. Finally, the relevance of these features to sense distinctions is assessed through qualitative evaluation.

Another type of classifier is trained for further investigation: *classifiers for WSD*. These are trained using the same sentence context features as input, but to directly resolve the ambiguity of each individual word. Unlike the classifiers for cluster assignment, which are trained to predict cluster labels, the classifiers for WSD are trained to predict sense labels. This helps investigate how useful sentence context features are for lexical ambiguity resolution in general, independently of the embeddings.

The results of the cluster explainability experiment reveal that different types of contextual features determine the similarity between the embeddings. These are function words, punctuation marks, content words in the sentences, position of the target word in the sentences, neighboring words, morphological properties, and the syntactic role of the target words. The qualitative analysis reveals an even more critical finding: the features that determine embedding similarity are mostly sense-irrelevant. These results confirm H2a and extend beyond it, revealing that sense-irrelevant features determine similarity in most cases, a stronger effect than expected.

To illustrate this, consider the verb ‘ask’. ‘ask’, in its first sense, means “to request something” as in (3-a) and in its second sense, means “to ask a question” as in (3-b).

- (3) ask:
- a. *to request something*: “He asked her for recommendation.”
  - b. *to ask a question*: “‘Who said that?’ he asked.”

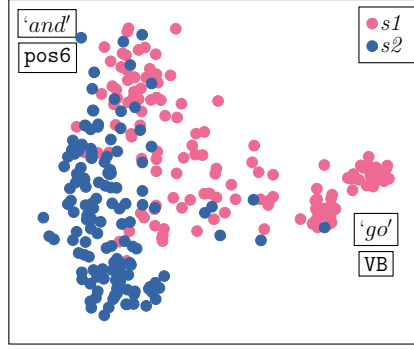


Figure 7.2: PCA visualization of the embeddings of ‘ask’ instances and important features found in each cluster. The features are ‘go’, ‘and’, VB (the verb is in base form), and pos6 (the word is the 6th token in the sentence).

Even though the sentence-feature classifier for WSD can successfully resolve the ambiguity of this verb ( $F1 = 0.81$ ), clustering achieves poor performance ( $ARI = 0.16$ ). The qualitative evaluation of the important features shows that the sentence context features that drive the cluster formation are sense-irrelevant. For instance, sentences in which the verb appears in base form (VB), sentences where the word is the sixth token (pos6), or sentences containing the words ‘go’ or ‘and’ are clustered together, and none of which are sense-relevant. The visualization of the embeddings of ‘ask’ instances and the important features per cluster, in Figure 7.2, further illustrates this.

**Homonymy-Polysemy Distinction.** The paper further assesses whether sense-irrelevant features affect embedding similarity differently across homonymous and polysemous senses due to their different levels of contextual overlap. Polysemous sense pairs have higher contextual overlap than homonymous senses due to their semantic relatedness. For instance, the senses of ‘bank’ (*financial institution* vs. *riverbank*) are semantically unrelated and tend to occur in clearly distinct contexts, as in (4), while the senses of ‘head’ (*body part* vs. *top part*) share meaning similarities and may occur in more similar contexts across senses, as in (5).

- (4) bank:
- a. I deposited money at the **bank**. SENSE: *financial institution*
  - b. I fished from the **bank**. SENSE: *riverbank*
- (5) head:
- a. I hit my **head** on the doorframe. SENSE: *body part*
  - b. I **head** of the hammer hit the floor. SENSE: *top part*

The results show that unsupervised WSD performance with contextualized word embeddings is higher for homonymous senses than polysemous senses ( $ARI$ : homonymy = 0.68, polysemy = 0.36). Furthermore, homonymous senses are more distinguishable by sentence context features; classifiers for WSD achieve better scores with homonymous senses ( $F1$ : homonymy = 0.80, polysemy = 0.72). This confirms that low contextual overlap with homonymy result in better ambiguity resolution performance. However, in relation to the effect of sense-irrelevant features, the qualitative analysis reveals that sense-irrelevant features determine embedding similarity for

both homonymy and polysemy. Even though homonymous senses are more distinguishable by sentence context features, the important features that drive cluster formation can still be sense-irrelevant for homonymous words.

This finding contradicts H2b: sense-irrelevant features determine embedding similarity across both ambiguity types, even in the case of low contextual overlap, i.e. when different senses occur in highly diverse contexts with clear cues. This is illustrated by the verb ‘ask’, which is homonymous in the senses given in (3). The classifier for WSD achieves good performance, confirming that its senses are distinguishable by sentence context features. However, clustering performs poorly, and the features driving cluster formation are sense-irrelevant. These findings suggest that sense-irrelevant features pose a fundamental challenge for unsupervised ambiguity resolution regardless of contextual overlap and lexical ambiguity type, explaining poor unsupervised performance and motivating the feature removal approach addressed in RQ3.

### 7.1.3 Removing Sense-Irrelevant Features (RQ3)

RQ3 is addressed in *Feature Removal with Adversarial Forgetting* (Yavas et al., 2024). RQ3 asks how we can successfully remove sense-irrelevant features from embeddings and what impact this has on unsupervised lexical ambiguity resolution performance. To address this question, the research paper *Feature Removal with Adversarial Forgetting* proposes an adversarial forgetting framework to remove sense-irrelevant features from the embeddings of PLMs. The paper specifically targets morphological and syntactic information based on the findings of the previous paper *Feature Relevance for WSD* (Yavas, 2024). The framework works by training a *forget-gate* on top of a PLM’s embeddings using an adversarial setup, aiming at removing the targeted information from the embeddings while preserving their word prediction capabilities.

With the proposed framework, a forget-gate is trained for each type of information: morphology and syntax. Two types of *feature-invariant embeddings* are created using these trained forget-gates from BERT embeddings: *morphology-* and *syntax-invariant*. The paper evaluates the success of the proposed method by using both word prediction and probing performance as measures. Both types of feature-invariant embeddings show low probing performance for the targeted unwanted information while preserving word prediction capabilities. These results demonstrate that the framework successfully removes the intended features without damaging the embeddings, confirming H3a.

Regarding lexical ambiguity resolution performance, the paper evaluates the impact of feature removal by comparing the performance of feature-invariant embeddings and the original embeddings on clustering-based Word Sense Induction (WSI). The results show that both types of invariant embeddings improve performance over the original BERT embeddings (ARI: BERT = 0.210, morphology-invariant = 0.232, syntax-invariant = 0.221), supporting H3b that removing sense-irrelevant features improves unsupervised lexical ambiguity resolution performance overall.

To illustrate this, consider the noun ‘area’. The BERT embeddings of its instances are clustered primarily by grammatical number (‘area’ vs. ‘areas’), even though there is no association between grammatical number and senses. This results in poor WSI performance (ARI = 0.03) driven by grammatical number.

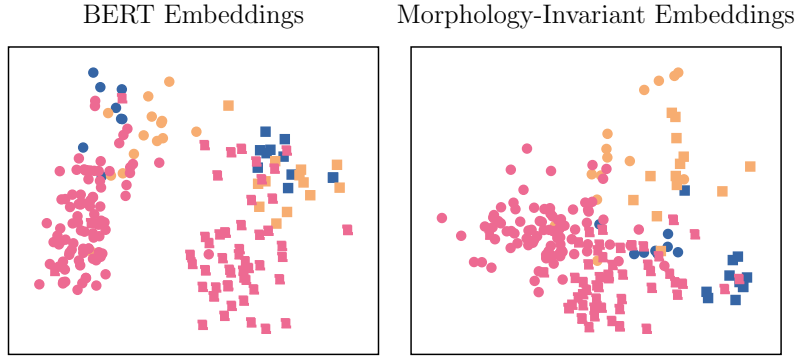


Figure 7.3: PCA visualizations of BERT embeddings (left) and morphology-invariant embeddings (right) of ‘area’ instances. Different data point colors refer to different senses. *Circles* refer to *singular nouns* and *squares* to *plural nouns*.

Morphology-invariant embeddings break this pattern, and the resulting clusters are more aligned with senses than with morphological features, leading to substantially better performance ( $\text{ARI} = 0.49$ ). This effect is further illustrated by the visualization of the embeddings and their senses and features in Figure 7.3.

Additionally, the paper provides a detailed analysis of the relation between morphological and syntactic information and WSI performance at the individual word level. While the overall results show that removing these features improves performance, this analysis investigates whether certain words benefit from retaining morphological or syntactic information, potentially revealing cases where removal causes harm. The results show that morphological information mostly introduces noise across individual words. In contrast, syntactic features show stronger association with senses than morphological features. However, even for words where syntactic features are associated with senses, removing this information causes only marginal performance drops.

To illustrate the negative effect of removing syntactic features, consider the noun ‘field’, whose BERT embeddings achieve good WSI performance ( $\text{ARI} = 0.62$ ), as syntactic role of the noun is associated with its senses. The third sense of ‘field’ (“somewhere where practical work is done or data is collected”) appears almost exclusively as the head of a compound in the data, as in (6).

- (6) field<sub>3</sub>:
- a. They will give suggestions that can be worked up into **field** *procedures*.
  - b. Actually, none of these papers says much directly about **field** *techniques*.

Syntax-invariant embeddings break this pattern, moving third sense instances closer to other sense instances and causing them to cluster with different senses, resulting in worse performance compared to the original BERT embeddings ( $\text{ARI} = 0.26$ ). However, despite differences at the individual word level, the performance gains from feature removal outweigh the losses overall. These findings confirm H3b that both feature types create more noise than signal overall, justifying their removal despite potential word-level variations.

### 7.1.4 Creating Sense-Enhanced Embeddings through Fine-Tuning (RQ4)

RQ4 is addressed in *Sense Knowledge Enhancement through Fine-Tuning* (Yavas et al., 2025). RQ4 asks whether successful sense-enhanced embeddings can be created through fine-tuning, how topological properties of the embeddings relate to task performance, and how different fine-tuning methods differ in their effect on the topology of the embedding space. To address these questions, the research paper *Sense Knowledge Enhancement through Fine-Tuning* compares SCL and SPL fine-tuning methods for sense knowledge enhancement, evaluating both methods with two PLMs: BERT and RoBERTa. The paper focuses on three topological properties: sense-alignment, uniformity, and isotropy. Task performance is evaluated on the Word-in-Context (WiC) task using both supervised and unsupervised setups. The paper further experiments with the temperature parameter of both loss functions to investigate its effect on topology and performance, providing a more complete picture of the relation between fine-tuning methods, topological properties, and downstream task performance.

The results reveal several key findings about the role of topological properties in the success of sense-enhanced embeddings. First, isotropy emerges as a critical property for performance. Embeddings with higher isotropy perform better. Improving the isotropy of contextualized word embeddings through post-processing alone, without incorporating additional sense knowledge, already leads to performance improvements. Second, increasing sense-alignment alone is insufficient for good performance: embeddings with only increased sense-alignment underperform compared to the original PLM embeddings. The best performance is achieved when both isotropy and sense-alignment are increased simultaneously and are both significantly high. Third, there is a trade-off between uniformity and sense-alignment, where optimal performance requires balancing these two properties. These findings partially confirm H4b. While the results support that improving isotropy alongside sense-alignment leads to better performance, uniformity should not be maximized but rather balanced with sense-alignment for optimal results. Additionally, the paper observes different patterns in the relation between topological properties and performance across unsupervised and supervised settings, showing that their relative importance may vary across settings. However, further investigation is needed to fully understand these differences.

Comparing the two fine-tuning methods, SPL and SCL produce embeddings that differ in terms of topological properties. More specifically, SCL generates embeddings that are more isotropic and more sense-aligned compared to SPL. However, uniformity values vary depending on the PLM. Together, these results support H4c: SCL and SPL differ in their effect on the topology of the embedding space, and SCL achieves a better balance for performance, as it creates embeddings that are more isotropic and more sense-aligned. Furthermore, SCL demonstrates superior and more consistent improvements across both models, while SPL fails to create successful embeddings with BERT. These findings show that successful sense-enhanced embeddings can be created through SCL but not SPL, partially confirming H4a.

### 7.1.5 Discussion and Synthesis

The five research papers in this thesis lead to a systematic understanding of how contextualized word embeddings encode and organize sense information. They further show that the identified limitations of the embeddings can be overcome with the proposed methods, leading to improvements in unsupervised lexical ambiguity resolution.

The research paper *Cross-Lingual Copredication Detection* (Yavas et al., 2023) (Chapter 2) shows that contextualized word embeddings encode sufficient information for detecting copredication and that this information transfers across languages in zero-shot settings. It further reveals a challenge in detecting the absence of copredication, attributed to the availability of both meaning facets with inherently polysemous nouns regardless of context. The research paper *Copredication Corpus Annotation* (Chen et al., 2025b) (Chapter 3) contextualizes the computational challenge by showing that copredication identification is challenging even for human experts. This challenge is reflected in low to moderate inter-annotator agreement. Furthermore, the analysis of annotator disagreements reveals that the disagreements stem from several factors, including unclear distinctions between the facets of inherently polysemous nouns. This confirms that determining the active facet of these nouns is also difficult for humans. Both papers together show that despite the complexity of the task, contextualized word embeddings can successfully be used for copredication detection in both monolingual and cross-lingual settings.

The research paper *Feature Relevance for WSD* (Yavas, 2024) (Chapter 4) identifies a core problem underlying poor unsupervised performance with contextualized word embeddings: these embeddings encode rich contextual information, and the features that determine embedding similarity between word instances are largely irrelevant to sense distinctions. This finding holds for both homonymy and polysemy. Even when the senses occur in highly diverse contexts and context provides clear cues for ambiguity resolution (low contextual overlap), such as in homonymy cases, the features that determine embedding similarity are still irrelevant to sense distinctions. This indicates that sense-irrelevant features pose a fundamental challenge for unsupervised ambiguity resolution regardless of the level of contextual overlap.

The research paper *Feature Removal with Adversarial Forgetting* (Yavas et al., 2024) (Chapter 5) provides a solution to this problem through the proposed adversarial forgetting framework and shows that selectively removing sense-irrelevant features (morphological and syntactic features) from contextualized word embeddings improves unsupervised performance. It further shows that even though these features can be sense-relevant for some words, they introduce more noise than signal and their removal benefits performance overall.

Finally, the research paper *Sense Knowledge Enhancement through Fine-Tuning* (Yavas et al., 2025) (Chapter 6) addresses the problem from the opposite direction and aims to incorporate sense knowledge into the embeddings through fine-tuning. It further addresses the question of whether improving sense-alignment through fine-tuning comes at the cost of embedding expressiveness and examines the effects of different fine-tuning methods on the topology of the embedding space. The findings reveal that successful sense-enhanced embeddings require more than just high sense-alignment; considering other topological properties such as isotropy and uniformity is also critical for task performance. Furthermore, the paper shows that the choice of fine-tuning method plays a crucial role, as different methods affect the embedding

space topology differently, resulting in different task performance. This highlights that how information is organized in the embedding space is a critical factor for unsupervised task performance, alongside what information is encoded.

Together, these findings identify two factors underlying poor unsupervised performance that stem from the nature of contextualized word embeddings. Although these embeddings encode sufficient information for resolving complex ambiguity phenomena, the unsupervised performance is limited by the influence of sense-irrelevant features in embedding similarity and by suboptimal topology. Furthermore, the papers presented in this thesis develop methods to overcome these limitations and the results show that these limitations can be addressed by improving existing PLM embeddings with these methods.

## 7.2 Contributions

### 7.2.1 Theoretical Contributions

**Embedding Properties Affecting Unsupervised Performance.** This thesis contributes to the understanding of properties that determine the effectiveness of contextualized word embeddings in unsupervised lexical ambiguity resolution tasks. We identify two key properties that affect unsupervised performance. First, embedding similarity can be determined by sense-irrelevant features, which is problematic given the central role of similarity in unsupervised settings. Second, topological properties of the embeddings affect task performance. Even with sense-aligned embedding spaces, topological properties such as isotropy play a critical role in performance, and their optimization leads to improvements.

**Contextual Features Driving Embedding Similarity.** While extensive research has investigated what information is encoded in contextualized word embeddings (Conneau et al., 2018; Tenney et al., 2019; Yenicelik et al., 2020; Zhao et al., 2020; Chersoni et al., 2021; Dalvi et al., 2022; Sajjad et al., 2022), which contextual features determine embedding similarity between same-word instances remains underexplored. Furthermore, while previous studies have observed that surface-level contextual and linguistic features affect embedding similarity (Wiedemann et al., 2019; Yenicelik et al., 2020; Gessler and Schneider, 2021; Laicher et al., 2021), this problem has not been systematically investigated. We address these gaps by systematically examining which contextual features determine embedding similarity.

**Embedding Similarity as a Proxy for Semantic Similarity.** The above investigation reveals that semantics-relevant information is not always the dominant factor determining embedding similarity. This finding has broader implications, as embedding similarity is commonly assumed to reflect semantic similarity. Our findings suggest that this equivalence cannot be assumed: the similarity relations may be determined by features irrelevant to semantic distinctions, potentially affecting downstream applications.

**Encoded Information and Unsupervised Task Performance.** We show that certain contextual information encoded in contextualized word embeddings can be

harmful for unsupervised downstream task performance. This highlights the need for better understanding of how encoded information relates to task performance. We address this gap in the context of lexical ambiguity resolution, demonstrating that feature relevance is critical for unsupervised performance.

**Fine-Tuning Methods and Embedding Space Topology.** In this thesis, we provide a detailed comparison of SPL and SCL fine-tuning methods in terms of their effect on embedding space topology. Our findings can explain the previously observed performance differences between SPL and SCL (Gunel et al., 2020; Khosla et al., 2020) by characterizing how each method affects the topology of the embedding space differently. Our investigation is one of the first systematic comparisons of SCL and SPL in the context of knowledge enhancement and provides insights for method selection.

**Topological Properties and Task Performance.** The importance of certain topological properties such as isotropy and uniformity for downstream task performance has been established in previous research (Biś et al., 2021; Gao et al., 2021; Liu et al., 2021a,b; Rajaei and Pilehvar, 2021). In this thesis, we extend this investigation to sense knowledge enhancement and lexical ambiguity resolution. We investigate how isotropy, sense-alignment, and uniformity relate to optimal performance.

## 7.2.2 Practical Contributions

**Method for Copredication Detection.** Despite receiving considerable attention from a theoretical perspective, copredication and inherently polysemous nouns have received limited attention from an empirical perspective. There was a need for a method that can automatically identify these constructions to support empirical research. In this thesis, we develop a supervised method for the automatic detection of copredication instances using contextualized word embeddings, which can serve as a practical tool for investigating this challenging phenomenon at scale.

**Method for Cluster Explainability.** While numerous approaches have been proposed to interpret contextualized word embeddings of PLMs (Conneau et al., 2018; Tenney et al., 2019; Yenicelik et al., 2020; Zhao et al., 2020; Chersoni et al., 2021; Dalvi et al., 2022; Sajjad et al., 2022, e.g.), the question of which contextual features determine embedding similarity has largely remained at the level of qualitative observation (Yenicelik et al., 2020; Gessler and Schneider, 2021; Laicher et al., 2021). In this thesis, we propose a systematic method to automatically identify the contextual features that determine similarity between contextualized word embeddings. This method can be used for the explainability of clustering-based applications by revealing which features drive clustering decisions. Furthermore, it can be extended to assess whether identified features are relevant to specific downstream tasks in different contexts.

**Methods for Improving Contextualized Word Embeddings:**

- **Adversarial Forgetting of Linguistic Information:** Adversarial forgetting has been widely applied in computer vision and, while adopted in Natural Language Processing (NLP), it has been mostly limited to the removal of domain-specific information (Louizos et al., 2017; Jaiswal et al., 2018; Peng and Zhang, 2020; Xin et al., 2022). In this thesis, we adapt this framework to target linguistic information encoded in contextualized word embeddings, resulting in embeddings that are invariant to targeted linguistic features. This method can be beneficial for research that aims to establish causal relations between specific information types and task performance and can be used to improve the embeddings for other tasks.
- **Supersense-Guided Fine-Tuning for Sense Knowledge Enhancement:** While existing work applies SCL for sense knowledge enhancement using word-level annotation (Mosolova et al., 2024), we propose two fine-tuning approaches that leverage supersense information using SCL and SPL. These methods enable general sense-alignment of the embedding space without relying on word-level, fine-grained sense annotations.

### 7.2.3 Resource Contributions

In addition to theoretical and methodological contributions, this thesis provides several publicly available resources to support future research on lexical ambiguity and contextualized word embeddings.

#### Datasets and Benchmarks:

- **Multilingual Copredication Test Sets:** Annotated test sets for copredication detection across five languages (Italian, English, German, Turkish, Chinese), enabling cross-lingual evaluation of this complex ambiguity phenomenon. (*Available at:* <https://github.com/yavasde/predication-classification>)
- **CoCoCorpus:** A gold-standard copredication corpus for English constructed through expert annotation. It supports research on inherent polysemy and copredication. (*Available at:* <https://github.com/CoCoCo-Project>)

#### Models and Tools:

- **Semantic Argument Type Classifiers:** Monolingual and multilingual classifiers for identifying semantic argument types of verbs and adjectives in context (for semantic types INFORMATION, EVENT, ARTIFACT, HUMAN, LOCATION). These models support copredication detection across languages. Furthermore, they can be applied to other research on argument structure, semantic types, or selectional preferences. (*Available at:* <https://github.com/yavasde/predication-classification>)
- **Forget-Gates for Feature Removal:** Forget-gate models for creating syntax-invariant and morphology-invariant embeddings from PLM embeddings by removing targeted linguistic features. (*Available at:* <https://github.com/yavasde/Adversarial-Forgetting-of-Morphosyntactic-Information>)

- **Sense-Enhanced Pre-Trained Transformer Language Models:** Sense-enhanced BERT and RoBERTa models. They support both research and practical applications in lexical ambiguity resolution. (*Available at Hugging Face:* <https://huggingface.co/yavasde>)
- **Cluster Explainability Tool:** A general-purpose tool for identifying sentence context features that determine similarity between contextualized word embeddings and lead to cluster formation. It supports explainability and feature relevance analysis across any clustering-based NLP task. (*Available at:* <https://github.com/yavasde/ContextDecode>)

**Code Repositories:** Implementations of all methods and experiments presented in this thesis are publicly available via GitHub repositories:

- **Cross-Lingual Copredication Detection:** <https://github.com/yavasde/predication-classification>
- **Feature Relevance for WSD:** <https://github.com/yavasde/ContextDecode>
- **Feature Removal with Adversarial Forgetting:** <https://github.com/yavasde/Adversarial-Forgetting-of-Morphosyntactic-Information>
- **Sense Knowledge Enhancement through Fine-Tuning:** <https://github.com/yavasde/Fine-Tuning-For-Sense-Enhancement>

## 7.3 Limitations

- **Model Coverage:** The experiments in this thesis are limited to BERT and RoBERTa. Since different PLMs differ in architecture, parameter size, training data, and training objectives, the findings may not generalize to other models.
- **Language Scope:** With the exception of the first research paper *Cross-Lingual Copredication Detection*, all experiments focus on English. Different languages may exhibit different patterns, and the extent to which the findings generalize across languages remains an open question.
- **Evaluation Scope:** Different research papers presented in this thesis employ different evaluation tasks based on their specific research questions. While this is methodologically appropriate, evaluating all methods across all tasks would enable more direct comparisons and better reveal task-specific effects.
- **Topological Properties:** The analysis of topological properties is limited to *sense-alignment*, *uniformity*, and *isotropy*. There may be other topological properties that influence the effectiveness of contextualized word embeddings in unsupervised lexical ambiguity resolution tasks.
- **Other Performance Factors:** This thesis focuses on information content and topology of the embedding space, but other model, data, or task-related factors can also affect the performance.

- **Resource Dependencies:** The experiments rely on WordNet, SemCor, and Typed Predicate Argument Structures (T-PAS), which may limit the generalizability of the findings to other sense inventories or datasets.

## 7.4 Future Research Directions

The findings of this thesis open several directions for future research. The adversarial forgetting method developed in this thesis successfully creates feature-invariant embeddings from contextualized word embeddings. However, given that isotropy emerged as a critical topological property related to task performance, future work could investigate whether optimizing isotropy during feature removal yields further improvements.

Additionally, future work could incorporate objectives for optimizing sense-alignment and other topological properties directly into the pre-training of models, rather than improving embeddings through fine-tuning. This could potentially achieve better performance and further validate the role of these properties in downstream task performance.

Furthermore, future work could investigate how topological properties relate to generalization capabilities of the embeddings. Specifically, examining whether embeddings with certain topological properties support better performance on unseen words, unseen senses, and out-of-domain data would provide insights into which properties lead to more generalizable sense-enhanced embeddings.

Another direction concerns the automation of feature relevance assessment. In this thesis, feature relevance assessment has been done through qualitative analysis, and feature removal decisions are based on this analysis. Future work could investigate methods for automated feature relevance assessment, eliminating the need for manual qualitative evaluation and enabling scalable application to diverse tasks and different types of information.

Finally, the insights gained through this thesis about the effects of feature relevance and topological properties on task performance could extend beyond lexical ambiguity resolution to other semantics-related tasks. Similar effects of non-semantic features and embedding expressiveness can be expected in other tasks, and uncovering these relations could lead to improvements across a broader range of NLP applications.

## 7.5 Concluding Remarks

Language is inherently ambiguous, and resolving ambiguity is fundamental for both language understanding and generation in NLP. Unsupervised lexical ambiguity resolution presents a particularly meaningful challenge. Supervised approaches rely on sense inventories and annotated data, but these resources are scarce and incomplete. They are available for only a limited number of languages and domains, and even where available, they cover a limited range of words and senses and quickly become outdated as language evolves. As a result, supervised approaches struggle to capture the full complexity of word meaning in real-world language use. This highlights the importance of unsupervised approaches, which have the potential to

enable systems that automatically adapt to new domains, languages, and emerging senses, making them more robust and scalable for real-world applications.

While contextualized word embeddings of PLMs are powerful representations, they show surprisingly limited performance on unsupervised lexical ambiguity resolution. Since unsupervised approaches depend mostly on the quality of the embeddings, a deeper understanding of what determines their effectiveness in these tasks is both theoretically important and practically necessary for overcoming their limitations. This thesis directly addresses this gap by investigating how these embeddings encode and organize sense information, and by developing methods to improve their effectiveness based on these insights.

The findings of this thesis reveal that what makes contextualized word embeddings effective for unsupervised lexical ambiguity resolution is more nuanced than whether they encode sufficient relevant information. Their effectiveness depends not only on whether relevant information is present, but also on whether irrelevant information introduces noise, and on how the information is organized in the embedding space. Overall, these findings advance our understanding of contextualized word embeddings, particularly regarding the properties that determine their effectiveness in unsupervised tasks. These findings also have broader implications for NLP and representation learning, as similar effects may be observed across a wider variety of tasks and encoder representations.

Beyond these theoretical insights, the findings of this thesis have practical value. The methods developed here demonstrate that the identified limitations can be addressed post-pretraining, without the need for training language models from scratch. This supports computational efficiency, reduced environmental impact, and the reusability of well-established models. More broadly, these findings highlight the value of a deeper theoretical understanding of existing models and demonstrate how such insights can directly inform practical improvements.

# Bibliography

- Ehsan Aghazadeh, Mohsen Fayyaz, and Yadollah Yaghoobzadeh. 2022. [Metaphors in pre-trained language models: Probing and generalization across datasets and languages](#). In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2037–2050, Dublin, Ireland. Association for Computational Linguistics.
- Eneko Agirre, Oier Lopez de Lacalle, Christiane Fellbaum, Shu-Kai Hsieh, Maurizio Tesconi, Monica Monachini, Piek Vossen, and Roxanne Segers. 2010. [SemEval-2010 task 17: All-words word sense disambiguation on a specific domain](#). In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 75–80, Uppsala, Sweden. Association for Computational Linguistics.
- Eneko Agirre and Aitor Soroa. 2007. [SemEval-2007 task 02: Evaluating word sense induction and discrimination systems](#). In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 7–12, Prague, Czech Republic. Association for Computational Linguistics.
- Jurij D. Apresjan. 1974. [Regular polysemy](#).
- Nicholas Asher. 2011. *Lexical meaning in context: A web of words*. Cambridge University Press.
- Marco Baroni, Silvia Bernardini, Adriano Ferraresi, and Eros Zanchetta. 2009. [The wacky wide web: a collection of very large linguistically processed web-crawled corpora](#). *Language Resources and Evaluation*, 43(3):209–226.
- Yonatan Belinkov, Lluís Màrquez, Hassan Sajjad, Nadir Durrani, Fahim Dalvi, and James Glass. 2017. [Evaluating layers of representation in neural machine translation on part-of-speech and semantic tagging tasks](#). In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1–10, Taipei, Taiwan. Asian Federation of Natural Language Processing.
- Geetanjali Bihani and Julia Rayz. 2021. [Low anisotropy sense retrofitting \(LASER\): Towards isotropic and sense enriched representations](#). In *Proceedings of Deep Learning Inside Out (DeeLIO): The 2nd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, pages 81–95, Online. Association for Computational Linguistics.
- Daniel Biś, Maksim Podkorytov, and Xiuwen Liu. 2021. [Too much in common: Shifting of embeddings in transformer language models and its implications](#). In

- Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5117–5130, Online. Association for Computational Linguistics.
- Terra Blevins and Luke Zettlemoyer. 2020. [Moving down the long tail of word sense disambiguation with gloss informed bi-encoders](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1006–1017, Online. Association for Computational Linguistics.
- Andreas Brocher, Stephani Foraker, and Jean-Pierre Koenig. 2016. [Processing of irregular polysemes in sentence reading](#). *Journal of experimental psychology. Learning, memory, and cognition*, 42 11:1798–1813.
- Alexander M Bronstein, Michael M Bronstein, and Ron Kimmel. 2003. Expression-invariant 3d face recognition. In *international conference on Audio-and video-based biometric person authentication*, pages 62–70. Springer.
- Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, and 1 others. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.
- Branden Chan, Stefan Schweter, and Timo Möller. 2020. [German’s next language model](#). In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6788–6796, Barcelona, Spain (Online). International Committee on Computational Linguistics.
- Long Chen, Laura Kallmeyer, and Rainer Osswald. 2022. [A frame-based model of inherent polysemy, copredication and argument coercion](#). In *Proceedings of the Workshop on Cognitive Aspects of the Lexicon*, pages 58–67, Taipei, Taiwan. Association for Computational Linguistics.
- Long Chen, Laura Kallmeyer, and Rainer Osswald. 2025a. Primary vs. secondary meaning facets of polysemous nouns. *Empirical issues in syntax and semantics: Selected papers from CSSP*, 2023:27.
- Long Chen, Deniz Ekin Yavaş, Laura Kallmeyer, and Rainer Osswald. 2025b. [Co-cocorpus: a corpus of copredication](#). In *Proceedings of the 21st Joint ACL - ISO Workshop on Interoperable Semantic Annotation (ISA-21)*, pages 31–40, Düsseldorf, Germany. Association for Computational Linguistics.
- Emmanuele Chersoni, Enrico Santus, Chu-Ren Huang, and Alessandro Lenci. 2021. [Decoding word embeddings with brain-based semantic features](#). *Computational Linguistics*, 47(3):663–698.
- Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. [Unsupervised cross-lingual representation learning at scale](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

- Alexis Conneau, German Kruszewski, Guillaume Lample, Loïc Barrault, and Marco Baroni. 2018. [What you can cram into a single \\$&!#\\* vector: Probing sentence embeddings for linguistic properties](#). In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2126–2136, Melbourne, Australia. Association for Computational Linguistics.
- D Alan Cruse. 1986. *Lexical semantics*. Cambridge university press.
- Fahim Dalvi, Abdul Rafae Khan, Firoj Alam, Nadir Durrani, Jia Xu, and Hassan Sajjad. 2022. [Discovering latent concepts learned in bert](#). *ArXiv*, abs/2205.07237.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [BERT: Pre-training of deep bidirectional transformers for language understanding](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
- Johannes Dölling. 2020. [Systematic polysemy](#). In Daniel Gutzmann, Lisa Matthewson, Cécile Meier, Hotze Rullmann, and Thomas Ede Zimmermann, editors, *The Wiley Blackwell Companion to Semantics*. Wiley Blackwell.
- Philip Edmonds and Scott Cotton. 2001. [SENSEVAL-2: Overview](#). In *Proceedings of SENSEVAL-2 Second International Workshop on Evaluating Word Sense Disambiguation Systems*, pages 1–5, Toulouse, France. Association for Computational Linguistics.
- Kawin Ethayarajh. 2019. [How contextual are contextualized word representations? Comparing the geometry of BERT, ELMo, and GPT-2 embeddings](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 55–65, Hong Kong, China. Association for Computational Linguistics.
- Christiane Fellbaum. 1998. *WordNet: An electronic lexical database*. MIT press.
- Jun Gao, Di He, Xu Tan, Tao Qin, Liwei Wang, and Tie-Yan Liu. 2019. [Representation degeneration problem in training natural language generation models](#). In *International Conference on Learning Representations*.
- Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. [SimCSE: Simple contrastive learning of sentence embeddings](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Marcos Garcia. 2021. [Exploring the representation of word meanings in context: A case study on homonymy and synonymy](#). In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3625–3640, Online. Association for Computational Linguistics.

- Aina Garí Soler and Marianna Apidianaki. 2020. [MULTISEM at SemEval-2020 task 3: Fine-tuning BERT for lexical meaning](#). In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 158–165, Barcelona (online). International Committee for Computational Linguistics.
- Luke Gessler and Nathan Schneider. 2021. [BERT has uncommon sense: Similarity ranking for word sense BERTology](#). In *Proceedings of the Fourth BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 539–547, Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Beliz Gunel, Jingfei Du, Alexis Conneau, and Ves Stoyanov. 2020. [Supervised contrastive learning for pre-trained language model fine-tuning](#). In *International conference on machine learning*.
- Janosch Haber and Massimo Poesio. 2020a. [Assessing polyseme sense similarity through co-predication acceptability and contextualised embedding distance](#). In *Proceedings of the Ninth Joint Conference on Lexical and Computational Semantics*, pages 114–124, Barcelona, Spain (Online). Association for Computational Linguistics.
- Janosch Haber and Massimo Poesio. 2020b. [Word sense distance in human similarity judgements and contextualised word embeddings](#). In *Proceedings of the Probability and Meaning Conference (PaM 2020)*, pages 128–145, Gothenburg. Association for Computational Linguistics.
- Janosch Haber and Massimo Poesio. 2021. [Patterns of polysemy and homonymy in contextualised language models](#). In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2663–2676, Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Zellig S Harris. 1954. Distributional structure. *Word*, 10(2-3):146–162.
- Sepp Hochreiter and Jürgen Schmidhuber. 1997. [Long short-term memory](#). *Neural Computation*, 9(8):1735–1780.
- Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. 2020. [XTREME: A massively multilingual multi-task benchmark for evaluating cross-lingual generalisation](#). In *Proceedings of the 37th International Conference on Machine Learning*, pages 4411–4421. Proceedings of Machine Learning Research.
- Linmei Hu, Zeyi Liu, Ziwang Zhao, Lei Hou, Liqiang Nie, and Juanzi Li. 2023. [A survey of knowledge enhanced pre-trained language models](#). *IEEE Transactions on Knowledge and Data Engineering*.
- Luyao Huang, Chi Sun, Xipeng Qiu, and Xuanjing Huang. 2019. [GlossBERT: BERT for word sense disambiguation with gloss knowledge](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3509–3514, Hong Kong, China. Association for Computational Linguistics.

- Ayush Jaiswal, Daniel Moyer, Greg Ver Steeg, Wael AbdAlmageed, and Premkumar Natarajan. 2020. Invariant representations through adversarial forgetting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 4272–4279.
- Ayush Jaiswal, Rex Yue Wu, Wael Abd-Almageed, and Prem Natarajan. 2018. Unsupervised adversarial invariance. *Advances in neural information processing systems*, 31.
- Ayush Jaiswal, Yue Wu, Wael AbdAlmageed, and Premkumar Natarajan. 2019. Unified adversarial invariance. *arXiv preprint arXiv:1905.03629*.
- Elisabetta Jezek, Bernardo Magnini, Anna Feltracco, Alessia Bianchini, and Octavian Popescu. 2014. T-PAS; a resource of typed predicate argument structures for linguistic analysis and semantic processing. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 890–895, Reykjavik, Iceland. European Language Resources Association (ELRA).
- Elisabetta Ježek and Chiara Melloni. 2009. Complex types in the (morphologically) complex lexicon. In *Proceedings of the 5th International Conference on Generative Approaches to the Lexicon*, pages 17–19.
- David Jurgens and Ioannis Klapaftis. 2013. [SemEval-2013 task 13: Word sense induction for graded and non-graded senses](#). In *Second Joint Conference on Lexical and Computational Semantics (\*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 290–299, Atlanta, Georgia, USA. Association for Computational Linguistics.
- Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. 2020. [Supervised contrastive learning](#). *Advances in neural information processing systems*, 33:18661–18673.
- Alireza Khotanzad and Yaw Hua Hong. 1990. Rotation invariant image recognition using features selected via a systematic method. *Pattern recognition*, 23(10):1089–1101.
- Ekaterini Klepousniotou. 2002. [The processing of lexical ambiguity: Homonymy and polysemy in the mental lexicon](#). *Brain and Language*, 81(1):205–223.
- Ekaterini Klepousniotou and Shari R. Baum. 2007. [Disambiguating the ambiguity advantage effect in word recognition: An advantage for polysemous but not homonymous words](#). *Journal of Neurolinguistics*, 20(1):1–24.
- Ekaterini Klepousniotou, G. Bruce Pike, Karsten Steinhauer, and Vincent Gracco. 2012. [Not all ambiguous words are created equal: An eeg investigation of homonymy and polysemy](#). *Brain and Language*, 123(1):11–21.
- Ekaterini Klepousniotou, Debra Titone, and Carolina Romero. 2008. Making sense of word senses: the comprehension of polysemy depends on sense overlap. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 34(6):1534.

- Alexandros Komninos and Suresh Manandhar. 2016. [Structured generative models of continuous features for word sense induction](#). In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3577–3587, Osaka, Japan. The COLING 2016 Organizing Committee.
- Henry Kučera, Winthrop Francis, William Freeman Twaddell, Mary Lois Marckworth, Laura M Bell, and John Bissell Carroll. 1967. Computational analysis of present-day american english. *Brown University Press, Providence, RI*.
- Severin Laicher, Sinan Kurtyigit, Dominik Schlechtweg, Jonas Kuhn, and Sabine Schulte im Walde. 2021. [Explaining and improving BERT performance on lexical semantic change detection](#). In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Student Research Workshop*, pages 192–202, Online. Association for Computational Linguistics.
- Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.
- Yoav Levine, Barak Lenz, Or Dagan, Ori Ram, Dan Padnos, Or Sharir, Shai Shalev-Shwartz, Amnon Shashua, and Yoav Shoham. 2020. [SenseBERT: Driving some sense into BERT](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4656–4667, Online. Association for Computational Linguistics.
- Yaobo Liang, Nan Duan, Yeyun Gong, Ning Wu, Fenfei Guo, Weizhen Qi, Ming Gong, Linjun Shou, Daxin Jiang, Guihong Cao, Xiaodong Fan, Ruofei Zhang, Rahul Agrawal, Edward Cui, Sining Wei, Taroon Bharti, Ying Qiao, Jiun-Hung Chen, Winnie Wu, and 5 others. 2020. [XGLUE: A new benchmark dataset for cross-lingual pre-training, understanding and generation](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6008–6018, Online. Association for Computational Linguistics.
- Bastien Liétard, Pascal Denis, and Mikaela Keller. 2024. [To word senses and beyond: Inducing concepts with contextualized language models](#). In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 2684–2696, Miami, Florida, USA. Association for Computational Linguistics.
- Fangyu Liu, Ivan Vulić, Anna Korhonen, and Nigel Collier. 2021a. [Fast, effective, and self-supervised: Transforming masked language models into universal lexical and sentence encoders](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1442–1459, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Nelson F. Liu, Matt Gardner, Yonatan Belinkov, Matthew E. Peters, and Noah A. Smith. 2019a. [Linguistic knowledge and transferability of contextual representations](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1073–1094, Minneapolis, Minnesota. Association for Computational Linguistics.

- Qianchu Liu, Fangyu Liu, Nigel Collier, Anna Korhonen, and Ivan Vulić. 2021b. [MirrorWiC: On eliciting word-in-context representations from pretrained language models](#). In *Proceedings of the 25th Conference on Computational Natural Language Learning*, pages 562–574, Online. Association for Computational Linguistics.
- Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019b. [Roberta: A robustly optimized bert pretraining approach](#). *arXiv preprint arXiv:1907.11692*.
- Christos Louizos, Kevin Swersky, Yujia Li, Max Welling, and Richard Zemel. 2015. The variational fair autoencoder. *arXiv preprint arXiv:1511.00830*.
- Christos Louizos, Kevin Swersky, Yujia Li, Max Welling, and Richard Zemel. 2017. The variational fair autoencoder.
- Daniel Loureiro and Alípio Jorge. 2019. [Language modelling makes sense: Propagating representations through WordNet for full-coverage word sense disambiguation](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5682–5691, Florence, Italy. Association for Computational Linguistics.
- Daniel Loureiro, Kiamehr Rezaee, Mohammad Taher Pilehvar, and Jose Camacho-Collados. 2021. [Analysis and evaluation of language models for word sense disambiguation](#). *Computational Linguistics*, 47(2):387–443.
- Suresh Manandhar, Ioannis Klapaftis, Dmitriy Dligach, and Sameer Pradhan. 2010. [SemEval-2010 task 14: Word sense induction & disambiguation](#). In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 63–68, Uppsala, Sweden. Association for Computational Linguistics.
- Federico Martelli, Najla Kalach, Gabriele Tola, and Roberto Navigli. 2021. [SemEval-2021 task 2: Multilingual and cross-lingual word-in-context disambiguation \(MCL-WiC\)](#). In *Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)*, pages 24–36, Online. Association for Computational Linguistics.
- Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.
- George A. Miller and Walter G. Charles. 1991. [Contextual correlates of semantic similarity](#). *Language and Cognitive Processes*, 6(1):1–28.
- George A. Miller, Claudia Leacock, Randee Teng, and Ross T. Bunker. 1993. [A semantic concordance](#). In *Human Language Technology: Proceedings of a Workshop Held at Plainsboro, New Jersey, March 21-24, 1993*.
- Andrea Moro and Roberto Navigli. 2015. [SemEval-2015 task 13: Multilingual all-words sense disambiguation and entity linking](#). In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 288–297, Denver, Colorado. Association for Computational Linguistics.

- Anna Mosolova, Marie Candito, and Carlos Ramisch. 2024. [Injecting Wiktionary to improve token-level contextual representations using contrastive learning](#). In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 34–41, St. Julian’s, Malta. Association for Computational Linguistics.
- Anna Mosolova, Marie Candito, and Carlos Ramisch. 2025. [In the LLM era, word sense induction remains unsolved](#). In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 17161–17178, Vienna, Austria. Association for Computational Linguistics.
- Jiaqi Mu and Pramod Viswanath. 2018. [All-but-the-top: Simple and effective post-processing for word representations](#). In *International Conference on Learning Representations*.
- Sathvik Nair, Mahesh Srinivasan, and Stephan Meylan. 2020. [Contextualized word embeddings encode aspects of human-like word sense knowledge](#). In *Proceedings of the Workshop on the Cognitive Aspects of the Lexicon*, pages 129–141, Online. Association for Computational Linguistics.
- Roberto Navigli. 2009. [Word sense disambiguation: A survey](#). *ACM Comput. Surv.*, 41(2).
- Roberto Navigli. 2012. A quick tour of word sense disambiguation, induction and related approaches. In *SOFSEM 2012: Theory and Practice of Computer Science*, pages 115–129, Berlin, Heidelberg. Springer Berlin Heidelberg.
- Roberto Navigli, David Jurgens, and Daniele Vannella. 2013. [SemEval-2013 task 12: Multilingual word sense disambiguation](#). In *Second Joint Conference on Lexical and Computational Semantics (\*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 222–231, Atlanta, Georgia, USA. Association for Computational Linguistics.
- Roberto Navigli and Daniele Vannella. 2013. [SemEval-2013 task 11: Word sense induction and disambiguation within an end-user application](#). In *Second Joint Conference on Lexical and Computational Semantics (\*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 193–201, Atlanta, Georgia, USA. Association for Computational Linguistics.
- Minlong Peng and Qi Zhang. 2020. [Weighed domain-invariant representation learning for cross-domain sentiment analysis](#). In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 251–265, Barcelona, Spain (Online). International Committee on Computational Linguistics.
- Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. [GloVe: Global vectors for word representation](#). In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

- Francesco Periti, David Alfter, and Nina Tahmasebi. 2024. [Automatically generated definitions and their utility for modeling word meaning](#). In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 14008–14026, Miami, Florida, USA. Association for Computational Linguistics.
- Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. [Deep contextualized word representations](#). In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.
- Matthew E. Peters, Mark Neumann, Robert Logan, Roy Schwartz, Vidur Joshi, Sameer Singh, and Noah A. Smith. 2019. [Knowledge enhanced contextual word representations](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 43–54, Hong Kong, China. Association for Computational Linguistics.
- Mohammad Taher Pilehvar and Jose Camacho-Collados. 2019. [WiC: the word-in-context dataset for evaluating context-sensitive meaning representations](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1267–1273, Minneapolis, Minnesota. Association for Computational Linguistics.
- Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. [How multilingual is multilingual BERT?](#) In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4996–5001, Florence, Italy. Association for Computational Linguistics.
- Sameer Pradhan, Edward Loper, Dmitriy Dligach, and Martha Palmer. 2007. [SemEval-2007 task-17: English lexical sample, SRL and all words](#). In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 87–92, Prague, Czech Republic. Association for Computational Linguistics.
- James Pustejovsky. 1995. *The Generative Lexicon*. MIT Press, Cambridge, MA.
- James Pustejovsky. 1998. [The semantics of lexical underspecification](#).
- James Pustejovsky and Elisabetta Jezek. 2012. [Semantic coercion in language: Beyond distributional analysis](#).
- Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, and 1 others. 2018. Improving language understanding by generative pre-training.
- Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. [Language models are unsupervised multitask learners](#).
- Alessandro Raganato, Iacer Calixto, Asahi Ushio, Jose Camacho-Collados, and Mohammad Taher Pilehvar. 2023. [SemEval-2023 task 1: Visual word sense disambiguation](#). In *Proceedings of the 17th International Workshop on Semantic*

- Evaluation (SemEval-2023)*, pages 2227–2234, Toronto, Canada. Association for Computational Linguistics.
- Alessandro Raganato, Jose Camacho-Collados, and Roberto Navigli. 2017. [Word sense disambiguation: A unified evaluation framework and empirical comparison](#). In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 99–110, Valencia, Spain. Association for Computational Linguistics.
- Sara Rajaei and Mohammad Taher Pilehvar. 2021. [A cluster-based approach for improving isotropy in contextual embedding space](#). In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 575–584, Online. Association for Computational Linguistics.
- Hassan Sajjad, Nadir Durrani, Fahim Dalvi, Firoj Alam, Abdul Khan, and Jia Xu. 2022. [Analyzing encoded concepts in transformer language models](#). In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3082–3101, Seattle, United States. Association for Computational Linguistics.
- Rita Sevastjanova, Aikaterini-Lida Kalouli, Christin Beck, Hanna Schäfer, and Menatallah El-Assady. 2021. [Explaining contextualization in language models using visual analytics](#). In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 464–476, Online. Association for Computational Linguistics.
- Benjamin Snyder and Martha Palmer. 2004. [The English all-words task](#). In *Proceedings of SENSEVAL-3, the Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text*, pages 41–43, Barcelona, Spain. Association for Computational Linguistics.
- Bayerische Staatsbibliothek and Stefan Schweter. 2025. [bert-base-italian-cased \(revision 843e404\)](#).
- Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. [BERT rediscovers the classical NLP pipeline](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601, Florence, Italy. Association for Computational Linguistics.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.
- Loïc Vial, Benjamin Lecouteux, and Didier Schwab. 2019. [Sense vocabulary compression through the semantic knowledge of WordNet for neural word sense disambiguation](#). In *Proceedings of the 10th Global Wordnet Conference*, pages 108–117, Wrocław, Poland. Global Wordnet Association.

- Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2019. Superglue: A stickier benchmark for general-purpose language understanding systems. *Advances in neural information processing systems*, 32.
- Feng Wang and Huaping Liu. 2021. [Understanding the Behaviour of Contrastive Loss](#). In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2495–2504, Los Alamitos, CA, USA. IEEE Computer Society.
- Tongzhou Wang and Phillip Isola. 2020. [Understanding contrastive representation learning through alignment and uniformity on the hypersphere](#). In *International conference on machine learning*, pages 9929–9939. PMLR.
- Gregor Wiedemann, Steffen Remus, Avi Chawla, and Chris Biemann. 2019. [Does bert make any sense? interpretable word sense disambiguation with contextualized embeddings](#). *ArXiv*, abs/1909.10430.
- Shijie Wu and Mark Dredze. 2020. [Are all languages created equal in multilingual BERT?](#) In *Proceedings of the 5th Workshop on Representation Learning for NLP*, pages 120–130, Online. Association for Computational Linguistics.
- Ji Xin, Chenyan Xiong, Ashwin Srinivasan, Ankita Sharma, Damien Jose, and Paul Bennett. 2022. [Zero-shot dense retrieval with momentum adversarial domain invariant representations](#). In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 4008–4020, Dublin, Ireland. Association for Computational Linguistics.
- Yadollah Yaghoobzadeh, Katharina Kann, T. J. Hazen, Eneko Agirre, and Hinrich Schütze. 2019. [Probing for semantic classes: Diagnosing the meaning content of word embeddings](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5740–5753, Florence, Italy. Association for Computational Linguistics.
- Yuanmeng Yan, Rumei Li, Sirui Wang, Fuzheng Zhang, Wei Wu, and Weiran Xu. 2021. [ConSERT: A contrastive framework for self-supervised sentence representation transfer](#). In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5065–5075, Online. Association for Computational Linguistics.
- Deniz Ekin Yavas. 2024. [Assessing the significance of encoded information in contextualized representations to word sense disambiguation](#). In *Proceedings of the Third Workshop on Understanding Implicit and Underspecified Language*, pages 42–53, Malta. Association for Computational Linguistics.
- Deniz Ekin Yavas, Timothée Bernard, Benoit Crabbé, and Laura Kallmeyer. 2025. [On the relation between fine-tuning, topological properties, and task performance in sense-enhanced embeddings](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 23610–23625, Vienna, Austria. Association for Computational Linguistics.

- Deniz Ekin Yavas, Timothée Bernard, Laura Kallmeyer, and Benoît Crabbé. 2024. [Improving word sense induction through adversarial forgetting of morphosyntactic information](#). In *Proceedings of the 13th Joint Conference on Lexical and Computational Semantics (\*SEM 2024)*, pages 238–251, Mexico City, Mexico. Association for Computational Linguistics.
- Deniz Ekin Yavas, Laura Kallmeyer, Rainer Osswald, Elisabetta Jezek, Marta Ricciardi, and Long Chen. 2023. [Identifying semantic argument types in predication and copredication contexts: A zero-shot cross-lingual approach](#). In *Proceedings of the 14th International Conference on Recent Advances in Natural Language Processing*, pages 310–320, Varna, Bulgaria. INCOMA Ltd., Shoumen, Bulgaria.
- David Yenicelek, Florian Schmidt, and Yannic Kilcher. 2020. [How does BERT capture semantics? a closer look at polysemous words](#). In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 156–162, Online. Association for Computational Linguistics.
- Haode Zhang, Haowen Liang, Yuwei Zhang, Liming Zhan, Xiaolei Lu, Albert Lam, and Xiao-Ming Wu. 2022. [Fine-tuning pre-trained language models for few-shot intent detection: Supervised pre-training and isotropization](#). In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 532–542, Seattle, United States. Association for Computational Linguistics.
- Mengjie Zhao, Philipp Dufter, Yadollah Yaghoobzadeh, and Hinrich Schütze. 2020. [Quantifying the contextualization of word representations with semantic class probing](#). In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1219–1234, Online. Association for Computational Linguistics.
- Zhi Zhong and Hwee Tou Ng. 2010. [It makes sense: A wide-coverage word sense disambiguation system for free text](#). In *Proceedings of the ACL 2010 System Demonstrations*, pages 78–83, Uppsala, Sweden. Association for Computational Linguistics.
- Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the IEEE international conference on computer vision*, pages 19–27.
- George Kingsley Zipf. 1945. The meaning-frequency relationship of words. *The Journal of general psychology*, 33(2):251–256.