University of Sheffield

Interpretable Computational Metaphor Processing

Shun Wang

Supervisor: Chenghua Lin

A report submitted in partial fulfilment of the requirements for the degree of PhD in Computer Science

in the

Department of Computer Science

January 7, 2026

Declaration

All sentences or passages quoted in this document from other people's work have been specifically acknowledged by clear cross-referencing to author, work and page(s). Any illustrations that are not the work of the author of this report have been used with the explicit permission of the originator and are specifically acknowledged. I understand that failure to do this amounts to plagiarism and will be considered grounds for failure.

Name: Shun Wang

Signature:Shun Wang

Date:05/03/2025

Acknowledgments

First and foremost, I would like to express my deepest gratitude to my supervisor, Prof. Chenghua Lin, for his invaluable guidance, encouragement, and unwavering support throughout my PhD journey. His insights, patience, and expertise have played a crucial role in shaping my research and academic growth. It has been an honor to work under his supervision.

I would like to acknowledge and thank the faculty members in the Department of Computer Science at the University of Sheffield, especially Dr. Loïc Barrault and Dr. Po Yang for their valuable advice and support during my PhD. Additionally, I would like to express my sincere gratitude to Prof. Han Wu from the School of Foreign Studies at the University of International Business and Economics. She provided me with invaluable opportunities to conduct cross-linguistic research and offered insightful guidance and support throughout my academic journey.

I would also like to extend my heartfelt gratitude to Dong Yang for his guidance and support during my internship at Baidu. His invaluable advice, technical insights, and mentorship helped me navigate the challenges of my research and broadened my understanding of real-world applications of natural language processing.

Furthermore, I am deeply appreciative of my research collaborators, Yucheng Li, Tyler Loakman, Ge Zhang, and Chen Tang, whose expertise and collaboration have greatly enriched my work. Their insights and discussions have been incredibly motivating.

My gratitude goes to my colleagues in the Natural Language Processing group for creating a supportive and stimulating research environment: Tomas Goldsack, Yizhi Li, Bohao Yang, Yiqi Liu, Chenghao Xiao, Xingwei Qu, Dr. Siwei Wu, Hanhua Hong, Yang Wang, William Thorne, Mali Jin, and Yida Mu. I would also like to thank my friends Kefan Chen, Changxu Chen, and Zechen Wang, for their invaluable discussions and wonderful companionship throughout this journey.

Finally, I would like to express my deepest gratitude to my family. In particular, I am profoundly thankful to my parents and my grandmother for their unwavering support and unconditional love. They have been my strongest pillars throughout

my life, always standing by my side through academic challenges and life's ups and downs. Most importantly, I want to extend my heartfelt gratitude to my wife and our beloved daughter. My wife has been my most steadfast companion, offering endless understanding, patience, and support as I pursued my academic aspirations. And my daughter—she is my greatest source of joy. Her laughter and pure curiosity remind me that life is not just about academic pursuits but also about love, growth, and cherishing the beautiful moments along the way. I am deeply grateful to my family, whose love and support have carried me to where I am today. This doctoral dissertation is not just my achievement; it is a shared milestone, a testament to the collective effort and dedication of our entire family.

Abstract

Metaphors are a fundamental component of human language, enabling abstract reasoning, nuanced communication, and cultural expression. However, their inherent complexity poses significant challenges for natural language processing (NLP) systems, which must accurately detect, interpret, and translate metaphorical expressions across diverse linguistic and cultural contexts. This thesis addresses these challenges by developing computational methodologies for interpretable metaphor processing, bridging insights from computational linguistics, psycholinguistics, and artificial intelligence.

The work advances three core areas: 1) **Enhancing metaphor detection** through syntactic pruning (RoPPT), semantic frame integration (FrameBERT), and basic meaning modeling (BasicBERT), achieving state-of-the-art performance across benchmark datasets. 2) **Improving cross-linguistic metaphor translation** via MMTE – a novel evaluation framework combining human and automatic metrics to assess emotional salience and translation quality in English, Chinese, and Italian. 3) **Leveraging interpretability in large language models (LLMs) for metaphor understanding** through sparse autoencoders and dictionary learning, decomposing latent representations to extract monosemantic features that improve metaphor transparency.

The findings contribute to the development of more linguistically sophisticated, contextually adaptive, and culturally aware NLP systems. By advancing metaphor processing methodologies and introducing interpretability techniques, this research provides a foundational exploration for applications in machine translation, sentiment analysis, and explainable AI. The proposed models and evaluation frameworks not only improve metaphor understanding in computational settings but also provide a

foundation for future work in cross-linguistic and cross-cultural NLP.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Metaphor is a foundational linguistic phenomenon that serves as both a cognitive framework and a powerful communicative device. By mapping abstract or unfamiliar concepts onto concrete and familiar domains, metaphors enable humans to make sense of complex ideas and express nuanced emotions, thereby enhancing the richness and versatility of language. Beyond their aesthetic appeal, metaphors are integral to human thought, influencing how we reason, make decisions, and interact with the world around us.

In the realm of computational linguistics, metaphor processing bridges the disciplines of psycholinguistics and artificial intelligence, offering a unique perspective to explore the interplay between language, cognition, and computation. This burgeoning field addresses the intricate challenges of detecting, interpreting, and even generating metaphors, tasks that demand an understanding of context, cultural nuances, and conceptual mappings. Its advancements have far-reaching implications for natural language processing (NLP), enhancing the depth and accuracy of tasks such as sentiment analysis, machine translation, overall interpretability.

1.1 Background and Motivation

Metaphorical expressions are a cornerstone of human language, appearing in nearly 20% of everyday communication (Steen 2010). They bridge abstract concepts and concrete experiences, making complex ideas relatable and accessible. For example, in the expression,

`This project is such a headache!`

the word *headache* metaphorically conveys a source of frustration or difficulty, diverging from its literal meaning of physical discomfort. This capacity for abstract mapping underscores metaphors' pivotal role in shaping human cognition, communication, and creativity (Lakoff & Johnson 1980, Lakoff 1993).

The study of metaphors is particularly compelling within computational linguistics, where it intersects with challenges in natural language understanding and generation. Metaphors, by their very nature, defy straightforward interpretation. They often depend on cultural and contextual cues, requiring nuanced comprehension. For instance, in the sentence,

`The scream pierced the night.`

the word *pierced* metaphorically conveys the sharp, penetrating quality of a sound, distinct from its literal association with physical penetration. Such examples highlight the multifaceted nature of metaphorical language, where meaning is derived from context-specific and conceptual mappings.

Despite their ubiquity, metaphors remain one of the most challenging aspects of language for computational models to process effectively. Their interpretation demands not only the recognition of figurative language but also an understanding of the broader conceptual structures that underpin metaphorical expressions. As a result, developing robust computational approaches to detect, interpret, and generate metaphors is essential for advancing NLP systems. Applications such as sentiment analysis, where metaphors often carry emotional undertones, or machine translation, which must navigate culturally specific metaphorical constructs, stand to benefit significantly from such advancements.

Understanding the intricate interplay between linguistic structure and metaphorical meaning is not merely an academic endeavor but a critical step toward creating AI systems that can truly grasp and replicate the richness of human language.

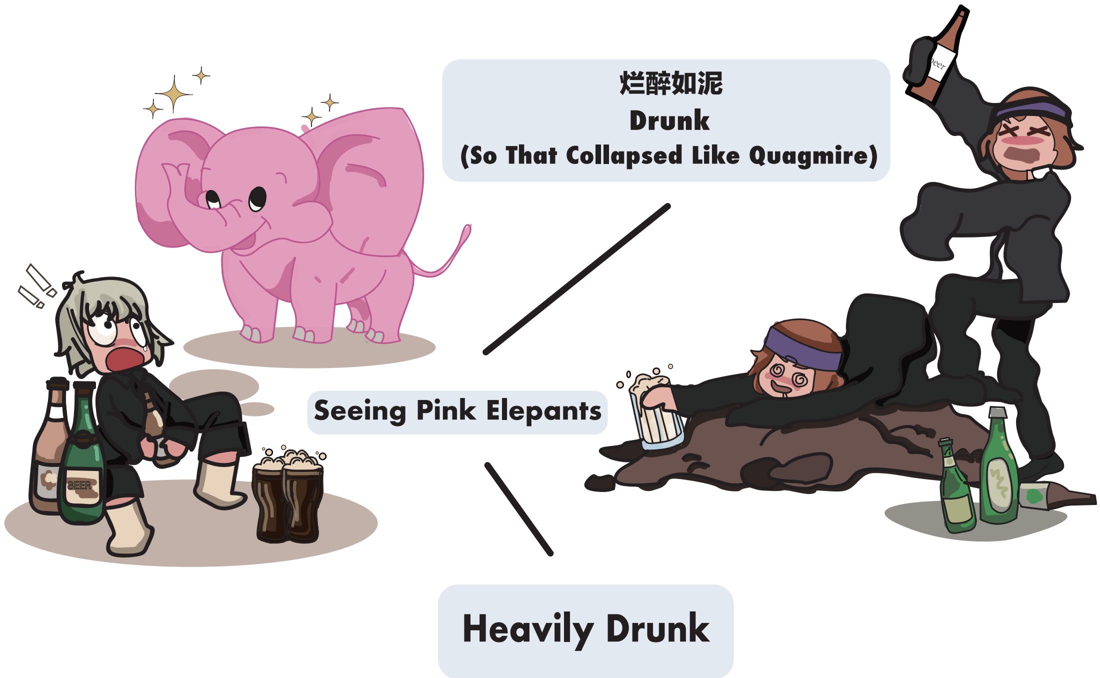


Figure 1.1: *Chinese and English metaphorical expressions of being drunk.*

1.1.1 Role of Metaphors in NLP

Metaphors are not mere linguistic flourishes; they are deeply embedded conceptual tools that profoundly shape human thought, communication, and decision-making (Lakoff & Johnson 1980). By linking abstract ideas with concrete experiences, metaphors enable individuals to structure and interpret complex phenomena, often influencing behaviors and societal norms. For instance, the conceptual metaphor "time is money" extends beyond language, shaping how societies prioritize efficiency, productivity, and resource allocation. Such metaphors encapsulate cultural values and shared cognitive frameworks, underscoring their significance beyond surface-level expression.

In the realm of computational linguistics, metaphors hold transformative potential for advancing natural language processing (NLP) systems. Unlike literal language, metaphorical expressions encapsulate layers of implicit meaning, requiring a sophisticated understanding of context, semantics, and cultural nuances. Successfully processing metaphors enables NLP systems to interpret emotional undertones and nuanced perspectives, which are critical in tasks like sentiment analysis. For example, the metaphor "a wave of grief" conveys an emotional intensity that goes beyond literal descriptions, providing richer insights into user sentiment (Abd Yusof et al. 2017).

Moreover, understanding metaphors is pivotal for tasks like machine translation, where figurative language often poses unique challenges. Direct translations of metaphors may not always capture their intended meaning due to cultural and linguistic differences. For example, while English speakers might describe someone as "seeing pink elephants" to denote drunkenness, a Chinese speaker might liken it to being "collapsed on the ground like quagmire." Computational systems equipped to recognize and translate such metaphors can bridge these linguistic and cultural gaps, enhancing cross-linguistic communication (Gutiérrez et al. 2017).

Beyond translation and sentiment analysis, metaphor processing has broader implications for fields like mental healthcare and creative AI applications. In therapeutic contexts, metaphors often serve as windows into a patient's mental state, offering valuable cues for diagnosis and treatment. For example, phrases like "drowning in sadness" or "shattered dreams" reflect deep emotional states that NLP systems could analyze to support mental health professionals. Similarly, creative applications such as story generation and dialogue systems benefit from incorporating metaphorical language, enabling them to produce more expressive and relatable content.

By addressing the intricate challenges of metaphor detection and interpretation, NLP research can unlock new dimensions of language understanding, ultimately creating systems that better emulate the richness and complexity of human communication.

1.2 Aims and Objectives

The overarching aim of this thesis is to advance the field of computational metaphor processing by addressing the multifaceted challenges associated with detecting, interpreting, and applying metaphors in diverse linguistic contexts. By bridging theoretical insights with practical innovations, this research seeks to contribute both foundational knowledge and applied methodologies that enhance the capabilities of natural language processing (NLP) systems. Specifically, the objectives of this thesis are as follows:

- Develop robust methodologies for detecting metaphorical expressions: This involves creating computational models capable of identifying metaphors in various linguistic and cultural contexts, ranging from conventional metaphors to more creative and context-dependent expressions. The focus is on achieving high accuracy and adaptability across multiple languages and domains.

- Explore techniques to denoise and refine contextual information for metaphor interpretation: Metaphor comprehension often hinges on context. This research aims to investigate advanced methods for filtering out irrelevant or noisy contextual elements while retaining and emphasizing semantically meaningful information. Techniques such as dependency parsing, target-oriented pruning, and attention mechanisms will be explored to enhance metaphor interpretation.

- Bridge cross-linguistic and cross-cultural gaps in metaphor processing: As metaphors are deeply influenced by cultural and linguistic factors, this research also aims to explore the universality and variability of metaphorical expressions. By analyzing cross-linguistic datasets and addressing challenges in metaphor translation, the thesis will contribute to building more culturally sensitive NLP systems.

- Enhance the explainability and interpretability of metaphor processing models: In line with the increasing demand for explainable AI, this thesis will also investigate approaches to make metaphor processing models more interpretable, enabling a better understanding of how they detect and interpret metaphorical language.

By addressing these objectives, this thesis aims to push the boundaries of what computational models can achieve in metaphor processing, paving the way for NLP systems that are not only more accurate but also more contextually aware, culturally adaptable, and semantically rich.

1.2.1 Research Questions

To fulfill the aims and objectives of this thesis, the following research questions are addressed, each designed to tackle critical aspects of computational metaphor processing and its integration into natural language processing (NLP) systems:

1. How can computational models effectively detect and interpret metaphorical expressions in context? This question focuses on developing models capable of identifying metaphors in diverse linguistic scenarios. It investigates how to leverage context, linguistic theories, and modern machine learning techniques to accurately detect metaphorical language. Additionally, it seeks to explore how these models can interpret the underlying conceptual mappings that give metaphors their meaning, ensuring that the systems are both precise and semantically insightful.

2. What techniques can improve the denoising of context for accurate metaphor understanding? Metaphor comprehension often requires isolating relevant contextual cues from noisy data. This question examines advanced techniques, such as syntactic pruning, dependency parsing, attention mechanisms, and representation learning, to filter out irrelevant information while preserving semantically significant elements. The goal is to enhance the model's ability to focus on the most informative aspects of the context, improving metaphor detection and interpretation accuracy.

3. What are the cross-linguistic and cultural challenges in metaphor processing, and how can they be addressed computationally? This additional question delves into the cultural and linguistic variations of metaphor usage, examining how computational models can effectively process and translate metaphors across different languages. It aims to identify universal patterns in metaphorical expressions and address challenges posed by culturally specific metaphors, contributing to more globally adaptable NLP systems.

4. How can metaphor processing models be made more interpretable and explainable? This question addresses the growing demand for explainable AI by investigating methods to enhance the interpretability of computational metaphor models. It examines how models can provide insights into their decision-making processes, particularly in detecting and interpreting metaphors, ensuring that the systems are transparent and trustworthy in their applications.

By answering these questions, this thesis aims to deepen our understanding of computational metaphor processing while contributing innovative solutions to enhance the capabilities of NLP systems.

1.3 Scope of the Study

This thesis is dedicated to advancing computational methodologies for the detection, interpretation, and translation of metaphorical expressions in natural language. Metaphors are an essential component of human communication, often carrying abstract, figurative meanings that differ from their literal interpretations. However, their inherent complexity poses significant challenges for natural language processing (NLP) systems, which must navigate diverse linguistic structures, cultural variations, and context-dependent conceptual mappings to accurately process metaphorical language. Addressing these challenges requires a multi-faceted approach that combines advances in machine learning, linguistic theory, and cognitive science to develop models that are both linguistically proficient and computationally efficient.

To tackle these challenges, this research explores three key aspects of metaphor processing: **Enhancing metaphor detection** – developing more precise and context-aware computational models to identify metaphorical expressions within natural language. **Improving metaphor translation** – ensuring that the figurative meaning of metaphors is preserved across different languages while accounting for cultural and structural differences. **Leveraging interpretability in large language models (LLMs) for metaphor understanding** – making LLMs more transparent and explainable in their handling of figurative language, improving both their usability and trustworthiness. By integrating insights from computational linguistics, psycholinguistics, and artificial intelligence, this thesis aims to construct robust and adaptable methodologies that improve the ability of NLP systems to detect, interpret, and translate metaphorical language with greater accuracy and contextual sensitivity. This research not only enhances the theoretical understanding of metaphor processing but also provides practical solutions for applications such as machine translation, sentiment analysis, and explainable AI, contributing to the development of NLP systems that are cognitively informed, linguistically sophisticated, and culturally aware.

1.3.1 Towards Robust Metaphor Detection

This chapter introduces three innovative approaches that address key challenges in computational metaphor detection. Each method brings unique contributions to improving the accuracy, interpretability, and applicability of metaphor processing models by leveraging advanced contextual, semantic, and syntactic techniques.

RoPPT: Target-Oriented Context Pruning for Metaphor Detection RoPPT (RoBERTa with Pruning on Target-Oriented Parse Tree) introduces a novel approach to metaphor detection by leveraging syntactic structures for effective context denoising. It reshapes dependency parse trees into flat, target-oriented structures, focusing on the semantically relevant neighbors of the metaphorical target. This pruning mechanism filters out irrelevant nodes, retaining only the essential contextual information for accurate metaphor detection. RoPPT addresses key challenges in computational efficiency and context relevance. Its streamlined tree structure facilitates batch optimization, overcoming the limitations of traditional dependency tree models. Experimental results on benchmark datasets (VUA, MOH-X, TroFi) demonstrate that RoPPT achieves superior performance compared to existing state-of-the-art models. Notably, the model's accuracy improves with sentence length, highlighting its robustness in handling complex linguistic structures.

FrameBERT: Conceptual Metaphor Detection with Frame Embedding Learning and Context Pruning FrameBERT is a BERT-based model designed to enhance metaphor detection by integrating external semantic knowledge from FrameNet. FrameNet provides a structured representation of conceptual frames, capturing relationships between entities and actions in a given context. FrameBERT leverages these embeddings to enrich the semantic representation of metaphorical expressions, addressing limitations of traditional models that rely solely on shallow contextual information. Additionally, FrameBERT employs a target-oriented context pruning mechanism that reshapes dependency parse trees into flat, target-focused structures. This approach filters out irrelevant context, enabling the model to focus on semantically significant elements surrounding the metaphorical target. Extensive evaluations on benchmark datasets (VUA, MOH-X, TroFi) demonstrate that FrameBERT significantly improves metaphor detection performance while enhancing model explainability by providing interpretable semantic frame representations.

Metaphor Detection via Explicit Basic Meanings Modelling This method introduces a novel mechanism called BasicMIP (Basic Metaphor Identification Process), which explicitly models the basic meanings of words to detect metaphors. Rooted in the Metaphor Identification Process (MIP) linguistic theory, BasicMIP contrasts a word's basic (literal) meaning with its contextual meaning to determine metaphorical usage. Unlike prior approaches that use aggregated or decontextualized embeddings (e.g., GloVe, RoBERTa) to approximate basic meanings, BasicMIP directly leverages basic annotations from the training data. By summing embeddings of instances labeled as literal, the model generates accurate basic meaning representations for each target word. These are then compared to contextual representations to identify semantic contrasts indicative of metaphors. The resulting model, BasicBERT, incorporates BasicMIP alongside a module for Selectional Preference Violation (SPV) and aggregated meaning modeling. Experimental results on VUA18 and VUA20 datasets show that BasicBERT outperforms state-of-the-art models, achieving F1 scores that approach theoretical upper bounds for annotated targets. BasicBERT's explicit modeling of basic meanings addresses critical gaps in metaphor detection while improving performance and interpretability.

These three approaches—RoPPT, FrameBERT, and BasicBERT—illustrate distinct yet complementary strategies for advancing metaphor detection: **RoPPT** focuses on syntactic relevance through target-oriented context pruning, ensuring efficient and precise metaphor detection. **FrameBERT** integrates external conceptual knowledge to enhance semantic richness and context relevance. **BasicBERT** employs explicit basic meaning modeling to refine the detection of semantic contrasts in metaphorical language. Together, these methods push the boundaries of computational metaphor processing, offering new insights and tools for tackling this inherently complex and context-dependent task.

1.3.2 Corpus and Metrics for Evaluating Machine Translation Quality of Metaphorical Language

Metaphors, deeply intertwined with language and culture, reflect how people from different linguistic backgrounds conceptualize abstract ideas through familiar terms. While metaphors often reveal universal cognitive patterns, their linguistic expressions vary significantly across cultures, shaped by distinct societal norms and worldviews. These variations create both opportunities and challenges for computational models tasked with processing metaphors in multilingual contexts.

These linguistic and cultural variations pose unique challenges for computational processing, particularly in tasks like machine translation. Translating metaphorical expressions requires more than direct word-to-word mapping; it demands a deep understanding of both the source and target cultures to preserve the intended meaning. Direct translations often result in semantic loss or misinterpretation, making it essential for computational models to account for the cultural and linguistic contexts that shape metaphors.

To address these challenges, this chapter explores the intricacies of metaphor translation using multilingual datasets, focusing on English, Chinese, and Italian. It examines how conceptual mappings, such as the mentioned "pierced the night" and its Chinese equivalent "穿透黑夜" (literally "penetrated the night"), retain their figurative meanings across languages despite structural and lexical differences. The study also investigates mismatches caused by linguistic norms, offering insights into how these challenges can be systematically approached through computational frameworks.

Special attention is given to the development of culturally aware computational models capable of detecting, interpreting, and translating metaphors with sensitivity to linguistic and cultural nuances. By integrating insights from cognitive linguistics, psycholinguistics, and computational techniques, this research proposes methods to enhance machine translation, cross-linguistic sentiment analysis, and other applications requiring a nuanced understanding of metaphorical language.

Additionally, this chapter introduces MMTE (Metaphorical Machine Translation Evaluation), a systematic framework for evaluating metaphor translation. MMTE contributes a manually annotated multilingual corpus between English, Chinese, and Italian, alongside a human evaluation framework to assess rhetorical equivalence. This framework bridges linguistic and computational insights, highlighting the challenges and solutions in metaphor translation.

Through its focus on cross-linguistic perspectives, this chapter contributes to a deeper understanding of how metaphors operate in different linguistic and cultural environments. It provides practical solutions for addressing the challenges of metaphor translation, paving the way for NLP systems that are not only linguistically proficient but also culturally adaptive and semantically nuanced.

1.3.3 Exploring Task Performance with Interpretable Models via Sparse Auto-Encoders

The rise of large language models (LLMs) such as GPT-4 and Llama 3 has brought transformative advancements to natural language processing, enabling systems to perform complex tasks with remarkable accuracy. However, the inherent opacity of these models presents significant challenges, particularly in specialized applications like metaphor detection and interpretation. Metaphors, characterized by their abstract, context-dependent meanings, demand nuanced understanding and precise reasoning capabilities that stretch the limits of current computational systems.

LLMs have shown potential for tackling metaphor processing, but their success hinges on addressing several key challenges. One major issue is the polysemantic nature of neurons in neural networks, where individual neurons respond to multiple, unrelated inputs. This phenomenon, driven by superposition, complicates the task of interpreting metaphorical expressions. For example, as the previous mentioned metaphor "*pierced the night*" requires the model to distinguish between the literal sense of "pierced" and its metaphorical connotation of a sharp, penetrating sound. Traditional approaches relying on external tools like POS tagging or knowledge graphs often fall short in capturing the depth of such abstract relationships.

This chapter leverages insights from mechanistic interpretability to address these challenges, applying techniques such as dictionary learning and sparse autoencoders. By decomposing LLMs into comprehensible components, this approach extracts monosemantic features from polysemantic neurons, reducing ambiguity and enabling the model to focus on contextually relevant information. This level of interpretability not only enhances metaphor detection but also provides transparency in how these models process abstract language.

Another critical insight from this research is the role of modular applications in enhancing LLM performance. By applying autoencoder-based feature decomposition in a task-specific manner, this chapter demonstrates significant improvements in metaphor detection and other abstract reasoning tasks. This modularity ensures that LLMs are not only capable of processing metaphors with higher accuracy but are also adaptable to various linguistic and cultural contexts.

In summary, processing metaphors with LLMs involves overcoming the opacity and polysemanticity inherent in these models. By employing advanced interpretability techniques and integrating them into task-specific workflows, this research contributes to a deeper understanding of how LLMs can handle complex, abstract reasoning. It supports the development of NLP systems that are both semantically rich and explainable, ensuring robust performance across structured and figurative language contexts.

1.4 Overview of the Thesis

This thesis explores the multifaceted challenges of computational metaphor detection and presents innovative approaches to advance the field. The research is structured into six main chapters, each addressing key aspects of metaphor processing, from foundational concepts to cutting-edge methodologies and cross-linguistic insights. The chapters are organized as follows:

- **Chapter 1: Introduction** This chapter provides an introduction to metaphor as a cognitive and linguistic phenomenon, highlighting its significance in natural language processing (NLP). It outlines the research objectives and defines the scope of the thesis. The chapter also highlights the key contributions made throughout the study.

- **Chapter 2: Literature Survey** This chapter reviews existing work in computational metaphor processing, covering theoretical frameworks, linguistic theories, and state-of-the-art methodologies. It delves into approaches for metaphor detection and metaphor interpretation, identifying gaps in the literature and motivating the development of the advanced methods proposed in this thesis.

- **Chapter 3: Towards Robust Metaphor Detection** This chapter introduces three novel approaches to metaphor detection:

  – RoPPT, a RoBERTa-based model that employs target-oriented parse tree pruning to denoise and refine the contextual information surrounding metaphorical targets.

  – FrameBERT, which integrates FrameNet embeddings and context pruning for conceptual metaphor detection.

  – BasicBERT, a model that explicitly incorporates basic meaning modeling to refine metaphor identification.

  Each method is described in detail, and experimental results are presented to demonstrate their effectiveness and contributions to metaphor processing.

- **Chapter 4: Corpus and Metrics for Evaluating Machine Translation Quality of Metaphorical Language** This chapter examines the role of linguistic and cultural variations in metaphorical expressions. By analyzing multilingual datasets, the chapter highlights universal conceptual patterns and culturally specific nuances. It also explores the challenges of metaphor translation and the importance of culturally aware computational models, with applications in machine translation and cross-linguistic sentiment analysis.

- **Chapter 5: Exploring Task Performance with Interpretable Models via Sparse Auto-Encoders** This chapter investigates how large language models (LLMs) such as GPT-4 and Llama3 can be leveraged for metaphor processing. It explores the interpretability of LLMs, their potential for handling abstract and context-dependent language, and the integration of advanced techniques like mechanistic interpretability and feature decomposition to enhance metaphor detection and interpretation. The chapter provides insights into how LLMs can be adapted for metaphor tasks while addressing their limitations.

- **Chapter 6: Conclusion and Future Work** The final chapter summarizes the key findings and contributions of the thesis. It reflects on the advancements made in metaphor detection and translation, emphasizing the significance of integrating semantic knowledge, context denoising, and cross-linguistic understanding. The chapter also outlines future research directions, including the potential for explainable AI in metaphor processing and the extension of proposed methods to other figurative language phenomena.

This structure ensures a logical progression from foundational concepts to advanced methodologies and practical applications. By addressing the theoretical, computational, and cultural dimensions of metaphor processing, this thesis contributes to the development of NLP systems that are not only more accurate but also more interpretable and adaptable across languages and domains.

1.5 Publications

This section presents a list of research publications that I have contributed to, focusing on metaphor detection, machine translation, and interpretable study. The works include peer-reviewed papers published in top conferences such as ACL, EMNLP, and EACL.

1.5.1 Published Works focussing on Metaphor

The following works have been published and focus on metaphor detection, evaluation and translation. These studies have been discussed and referenced throughout this thesis, providing essential background and contributions to the research presented.

In the first paper, we proposed a cross-linguistic framework for evaluating metaphor translation quality by annotating and analyzing multilingual datasets that reflect both universal and culture-specific metaphorical patterns. I was primarily responsible for all experimental work, including data collection, annotation organization, and statistical analysis, as well as drafting the main body of the paper.

In total, Papers 2, 3, and 4 are three closely related works that represent our initial exploration on metaphor detection. The second work, in which I am listed as a co-first author (first in order), was primarily led by me, with full responsibility for the design and execution of the experiments, as well as the writing of the manuscript. My co-authors played a key role in the later stages of the paper revision.

For the third and fourth works, I am listed as a co-first author (second in order). These works build upon our earlier research and share many common ideas, developed jointly through extensive discussions with my co-authors and substantial conceptual guidance from my supervisor, Chenghua Lin. In these works, I led the design and execution of all experiments, ensuring methodological consistency with our previous research. The writing was primarily completed by my co-authors, while I actively participated in revising and refining the manuscripts. The findings of this joint work have been fully integrated into this thesis and extended with additional analyses and experiments.

1. **Wang, S.**, Zhang, G., Wu, H., Loakman, T., Huang, W., Lin, C. *MMTE: Corpus and metrics for evaluating machine translation quality of metaphorical language.* In Proceedings of the Conference on Empirical Methods in Natural

Language Processing (**EMNLP**), 2024.

2. **Wang, S.***, Li, Y.*, Lin, C., Barrault, Loic., Guerin, F. *Metaphor detection with effective context denoising.* The European Chapter of the ACL (**EACL**), 2023.

3. Li, Y.*, **Wang, S.***, Lin, C., Guerin, F., Barrault, Loic. *FrameBERT: Conceptual metaphor detection with frame embedding learning.* The European Chapter of the ACL (**EACL**), 2023.

4. Li, Y.*, **Wang, S.***, Lin, C., Guerin, F. *Metaphor detection via explicit basic meanings modelling.* ACL Association for Computational Linguistics (**ACL**), 2023.

1.5.2 Under Review and other Works

This subsection includes ongoing research and papers under review. Notably, it covers work on interpretable models using sparse autoencoders(mentioned above) and biomedical summarization with knowledge aggregation. For Paper 5, I was responsible for all experimental work as well as drafting the main body of the manuscript. For Paper 6, I contributed to data preprocessing for the experiments and was responsible for revising the manuscript. These studies aim to enhance transparency in AI models and improve information synthesis in specialized domains.

5. **Wang, S.**, Loakman, T., Lei, Y., Liu, Y., Yang, B., Zhao, Y., Yang, D., Lin, C. *Exploring Task Performance with Interpretable Models via Sparse Auto-Encoders.* (**under review**).

6. Tang, C., **Wang, S.**, Goldsack, T., Lin, C. *Improving biomedical abstractive summarisation with knowledge aggregation from citation papers.* In Proceedings of the Conference on Empirical Methods in Natural Language Processing (**EMNLP**), 2023.

[0] * The two authors contributed equally to the works.

Chapter 2

Literature Survey

The computational study of metaphors represents a multidisciplinary effort bridging psycholinguistics, cognitive science, and artificial intelligence. Metaphors are a cornerstone of human communication, enabling us to express abstract and complex ideas through figurative language. They permeate every aspect of human discourse, from casual conversations to literary masterpieces, serving not merely as stylistic devices but as tools for cognitive and conceptual framing. Understanding and processing metaphors computationally has profound implications for enhancing natural language processing (NLP) systems, making them more effective in tasks such as sentiment analysis, machine translation, dialogue generation, and creative text synthesis.

Despite their ubiquity in human communication, metaphors pose significant challenges for computational systems due to their abstract and context-dependent nature. While literal language operates within the boundaries of explicit meaning, metaphorical language transcends these boundaries, relying heavily on shared knowledge, cultural context, and conceptual mappings. The difficulty lies in enabling machines to navigate this figurative terrain with precision and interpretive depth.

This chapter delves into the theoretical foundations, computational advancements, and persistent challenges in metaphor research. It highlights the role of metaphor in cognition and communication, surveys the key methodologies for metaphor detection and interpretation, and underscores the significance of cross-cultural variability in metaphor usage. These explorations lay the groundwork for the novel methodologies and contributions presented in this thesis, which aim to advance the field of computational metaphor processing.

2.1 Theoretical Foundations of Metaphor

Metaphors are figurative expressions involving one or more words that create semantic contrast between their contextual and basic meanings, as defined by Pragglejaz (2007). The contextual meaning is shaped by the surrounding context, including culture, sentiment, and situational factors, while the basic meaning is typically more concrete, body-related, and can be traced back to the word's original sense and earliest attested usage (Pragglejaz 2007). Beyond this contrast, metaphor processing also involves both contrast-based and similarity-based mappings, reflecting the cognitive mechanisms that allow speakers to relate disparate conceptual domains (Barnden 2021). Moreover, research in Cognitive Linguistics and psychological experimentation suggests that the mind actively engages in source–target metaphorical mappings even in the absence of overt metaphorical language, indicating that metaphor is not merely a rhetorical device but a fundamental organizing structure of conceptual thought (Barnden 2020). This dual-layered semantic structure, combined with these underlying cognitive mappings, underscores the complexity of metaphors as linguistic constructs that straddle the literal and the abstract.

This foundational definition establishes a vital bridge between classical linguistic frameworks and modern cognitive theories, particularly Conceptual Metaphor Theory (CMT) (Lakoff & Johnson 1980). While classical perspectives focus on the structural and functional aspects of metaphor as a linguistic phenomenon, cognitive theories like CMT emphasize the underlying mental processes that give rise to metaphorical expressions. CMT posits that metaphors are not merely decorative or stylistic devices but are deeply rooted in our cognitive architecture, shaping how humans perceive, reason, and communicate abstract ideas. This synergy between linguistic and cognitive approaches highlights the complementary nature of metaphor analysis, combining structural precision with conceptual depth.

In this section, we survey the theoretical foundations of metaphor by synthesizing linguistic and cognitive theories with cultural perspectives. The goal is to provide a comprehensive framework for understanding the nature of metaphors, their role in communication, and their implications for broader fields such as psycholinguistics and computational linguistics. Subsequent Chapters will build on these foundations, examining specific methodologies and applications in detail.

2.1.1 Selectional Preference Violation

Selectional Preference Violation (SPV), introduced by Wilks (1975, 1978), provides a semantic framework for analyzing metaphors by focusing on deviations from expected word pair associations. The theory posits that metaphors emerge when the selectional preferences inherent in a predicate's typical usage are violated by its arguments. This semantic anomaly signals a figurative interpretation, making SPV a valuable tool for understanding how metaphors function within linguistic structures.

Understanding SPV in Metaphors

Selectional preferences are implicit semantic constraints that determine which arguments are acceptable for a predicate. For example:

`The dog` ***`drinks`*** `water.`

`The car` ***`drinks`*** `gasoline.`

the verb ***`drink`*** typically aligns with an animate subject (e.g., `dog`) and a liquid object (e.g., `water`). When these preferences are violated, as in the metaphor '`The car` ***`drinks`*** `gasoline`', a figurative interpretation is required to make sense of the phrase. Here:

Car does not belong to the animate subject category.

Gasoline does not belong to the drinkable liquid category.

this double violation of selectional preferences transforms a literal interpretation into a metaphorical one, suggesting a conceptual mapping where the car is likened to an animate being consuming fuel.

SPV operates bidirectionally, examining both the target word (drink) and its contextual elements (car and gasoline). This approach allows SPV to identify the semantic incongruity that gives rise to metaphors. Researchers have leveraged SPV to design models that compare target words with their contextual information, improving metaphor detection systems (Mao et al. 2019, Choi et al. 2021, Su, Wu & Chen 2021).

Applications of SPV Beyond Metaphor Analysis

SPV has also been applied to various natural language processing (NLP) tasks, demonstrating its versatility:

- Word Sense Disambiguation: Resolving ambiguities in word meanings by analyzing context-specific preferences (Gallant 1991, Agirre & Stevenson 2007).
- Named Entity Recognition: Identifying named entities in text by examining selectional preferences (Ratinov & Roth 2009).
- Pronoun Resolution: Disambiguating pronoun referents by aligning them with compatible antecedents (Bergsma et al. 2008).
- Textual Inference: Assessing logical relationships between textual statements (Ritter et al. 2010).

These applications highlight SPV's broader relevance in understanding and modeling language beyond metaphor analysis.

Limitations of SPV in Metaphor Detection

While SPV is effective in identifying novel metaphors, it encounters challenges in detecting conventional metaphors—those deeply embedded in cultural and linguistic norms. According to Lakoff and Johnson (1980): **Conventional metaphors** are integrated into the ordinary conceptual system of a culture and are reflected in everyday language. For example, *spending time* is a metaphorical expression but is so common that it aligns with conventional selectional preferences. **Novel metaphors**, by contrast, introduce new conceptual mappings, often challenging existing linguistic norms.

In corpus-based studies, conventional metaphors may exhibit co-occurrence frequencies that surpass those of their literal counterparts. For instance, the collocation <spend, object, time>is more frequent in the British National Corpus than <spend, object, money>, making it difficult for SPV to distinguish the metaphorical nature of the former.

Additionally, SPV can mistakenly classify incorrect collocations as metaphors. For example, the phrase `My car` ***`reads`*** `gasoline` violates selectional preferences but lacks figurative coherence, making it a meaningless sentence rather than a metaphor. This demonstrates SPV's vulnerability to false positives and underscores the need for mechanisms to filter out such cases.

Overall, Selectional Preference Violation provides a different semantic method for analyzing metaphors, emphasizing the role of contextual incongruity in figurative language. Its applications extend beyond metaphor detection, contributing to key NLP tasks such as word sense disambiguation and named entity recognition. Nevertheless, the limitation outlined above indicate that SPV alone may not suffice. Further exploration of complementary or alternative approaches is needed to address these limitations and enhance metaphor identification in computational systems.

2.1.2 Conceptual metaphor theory

Conceptual Metaphor Theory (CMT), introduced by Lakoff and Johnson in ***Metaphors We Live By*** (1980), revolutionized the understanding of metaphors by positioning them as fundamental cognitive tools rather than mere linguistic flourishes. CMT posits that metaphors are rooted in cross-domain mappings between a source domain (typically concrete and grounded in sensory experiences) and a target domain (often abstract and conceptual). These mappings are not arbitrary; they arise from embodied experiences and serve as cognitive mechanisms that help humans navigate complex or abstract concepts by grounding them in familiar, tangible domains.

Metaphors as Cognitive Processes

Lakoff and Johnson argued that metaphors reveal distinct cognitive processes, enabling people to conceptualize and reason about abstract ideas through concrete experiences. For example:

`She` ***`attacked`*** `his argument.`

the metaphor *ARGUMENT IS WAR* demonstrates how the structure of conflict (a **source domain**) informs the way arguments (a **target domain**) are conceptualized. This mapping is evident in expressions like:

`I have never` ***`won`*** `an argument with her.`

`He` ***`shot down`*** `all of my arguments.`

`Your claims are` ***`indefensible`***`.`

In these examples, *WAR*-related concepts such as aggression, defense, and victory are transferred to the domain of *ARGUMENT*. Importantly, Lakoff and Johnson observed that such mappings often occur unconsciously, highlighting the deep integration of metaphor in human thought.

Beyond linguistic expressions, the cognitive underpinnings of metaphors extend to broader domains of creativity and problem-solving. By leveraging the properties or relationships inherent in source domains, metaphors introduce novelty and innovation in language and thought. This creative aspect underscores why metaphors are pervasive not only in communication but also in artistic, scientific, and technical domains.

Source and Target Domains in Metaphor

CMT categorizes metaphors into source and target domains, which play distinct roles in metaphorical reasoning. Lakoff and Johnson provided the following definition:

> *"In a metaphor, there are two domains: the target domain, which is constituted by the immediate subject matter, and the source domain, in which important metaphorical reasoning takes place and that provides the source concepts used in that reasoning."*

For example, in the metaphor *ARGUMENT IS WAR*, *ARGUMENT* is the target domain (abstract and complex), and *WAR* is the source domain (concrete and familiar). The source domain contributes properties such as aggression, strategy, and cooperation, which shape the understanding of the target domain. Lakoff (1993) extended the concept to include common source-target domain pairings, such as:

TIME PASSING IS MOTION: `The time has long since` ***`gone`*** `when...`

LIFE IS A JOURNEY: `He's` ***`gone through`*** `a lot in life.`

MAKING PROGRESS IS FORWARD MOVEMENT: `We are` ***`moving ahead`***`.`

These mappings illustrate how metaphors structure not only language but also fundamental human cognition.

Challenges and Limitations of CMT

While CMT provides a framework for understanding metaphors, it faces several challenges, particularly in formalizing and standardizing source-target mappings:

Subjectivity in Concept Representation: The identification and abstraction of source and target domains often rely on the interpreter's judgment. For example, in the metaphor argument is war, one might question whether battle or conflict would serve as more precise source domains. Such variability introduces ambiguity into the analysis.

Lack of Standardized Abstraction: Determining the appropriate level of abstraction for concepts in source and target domains remains an open question. For instance, while war may emphasize aggression and strategy, a more generalized term like conflict might dilute these nuances, failing to capture the metaphor's richness.

Computational Modeling Challenges: Representing and processing conceptual mappings computationally is non-trivial. The fluid and context-dependent nature of metaphors complicates the development of algorithms capable of identifying and modeling these mappings effectively.

Broader Impacts of CMT

CMT's influence extends beyond linguistics, shaping research in psychology, cognitive science, and computational linguistics. It has provided a foundation for exploring how metaphors:

- Structure mental models and influence decision-making (Osbeck et al. 2010, Tileagă 2013).
- Facilitate creativity and innovation in language (Brinton & Brinton 2010).
- Enhance computational applications such as metaphor detection and generation (Mason 2004, Ge et al. 2022).

Moreover, CMT has inspired research into how metaphors vary across cultures, reflecting diverse conceptualizations of shared experiences. For example, the English metaphor *time is money* emphasizes efficiency and economic value. This metaphor initially came into English about the time of the industrial revolution, when people started to be paid for work by the amount of time they worked. However, other cultures may conceptualize time through metaphors that prioritize relationships or nature. For example, in Chinese, the expression 逝者如斯夫 describes 'time passing by like running water'. CMT provides a perspective through which to examine how metaphors bridge the gap between language and thought. By revealing the deep cognitive structures underpinning metaphorical expressions, CMT not only enriches our understanding of communication but also offers practical insights for fields ranging from education to artificial intelligence.

Conceptual Metaphor Theory fundamentally redefines metaphors as cognitive phenomena rather than decorative linguistic devices. By elucidating the cross-domain mappings between source and target domains, CMT provides a framework for understanding how metaphors shape thought, language, and culture. However, the theory's reliance on subjective interpretation and its challenges in computational application highlight the need for continued refinement.

2.1.3 Metaphor Identification Procedure

The Metaphor Identification Procedure (MIP) was introduced by Pragglejaz (2007) as a standardized framework for identifying metaphors in texts. This approach was designed to address the subjectivity often associated with metaphor annotation in large corpora. Building on this foundation, Steen (2010) refined MIP into the Metaphor Identification Procedure Vrije Universiteit (MIPVU) and applied it to the creation of the VU Amsterdam Metaphor Corpus (VUA), the largest token-level metaphor identification dataset to date. These methodological advancements provide a systematic and practical guideline for understanding metaphors, bridging linguistic theories with computational applications.

Steps of MIP

MIP provides a clear, step-by-step process to identify metaphors in text at the lexical level:

1. **Read the Text-Discourse**: Develop a comprehensive understanding of the overall meaning and context.

2. **Identify Lexical Units**: Break down the text into discrete lexical units (e.g., words, phrases).

3. **Determine Contextual Meaning**: Establish the meaning of each lexical unit within its immediate context, taking into account its relationships with other words in the text.

4. **Identify Basic Meaning**: For each lexical unit, determine whether it has a more basic contemporary meaning in other contexts. Basic meanings are typically: a) More concrete: Evoking physical experiences like seeing, hearing, or touching. b) Body-related: Connected to physical actions or sensations. c) More precise: Opposed to abstract or vague interpretations. d) Historically older: Reflecting earlier or foundational meanings.

5. **Compare Contextual and Basic Meanings**: If the contextual meaning

contrasts with the basic meaning but can still be understood in relation to it, mark the lexical unit as metaphorical.

Consider the sentence: Fear clogged his mind. Step 1: **General Understanding** The sentence conveys that fear overwhelmed the subject's mental faculties. Step 2: **Lexical Units** Identify the units: fear, clogged, his, and mind. Step 3: **Contextual and Basic Meaning Analysis**: Fear: The contextual meaning is an intense emotion caused by danger or anxiety. The basic meaning is similar, so it is literal. Clogged: The contextual meaning refers to fear obstructing mental function, likening the mind to a pipe or passage. The basic meaning, "to block a pipe or passage," contrasts with the contextual meaning, making it metaphorical. His: Both the contextual and basic meanings indicate possession, so it is literal. Mind: The contextual and basic meanings—relating to the cognitive faculties—are similar, so it is literal. Step 4: **Conclusion** In this analysis, clogged is identified as metaphorical, while the other words are literal.

MIP and Computational Models

MIP has inspired numerous computational approaches for metaphor detection. Studies such as those by Song et al. (2021), Lin et al. (2021), Ottolina et al. (2021), and Qin and Zhao (2021) demonstrate the utility of MIP in guiding algorithm design, particularly in token-level metaphor identification.

While MIP aligns with Selectional Preference Violation (SPV) in its focus on contextual contrasts, it differs in its reliance on explicit definitions of basic and contextual meanings rather than frequency-based word pair associations. MIP also provides a more structured and systematic framework for metaphor annotation compared to the broader conceptual approach of Conceptual Metaphor Theory (CMT).

Limitations of MIP

Despite its strengths, the Metaphor Identification Procedure (MIP) is not without challenges. These limitations highlight the complexity of metaphor annotation and the areas where further refinement is needed to improve its consistency and applicability.

Ambiguity in Basic Meanings Words with multiple meanings often lack clear and universally agreed-upon distinctions between their basic and contextual meanings. For instance, the word *see* can mean "to perceive with the eyes" (a literal interpretation) or "to understand" (an abstract sense). Determining which meaning is more basic depends on subjective judgment, as definitions may vary across dictionaries and cultural contexts. This ambiguity can lead to disagreements among annotators, especially when the contextual meaning is nuanced or overlaps significantly with multiple potential basic meanings.

Subjectivity in Annotation The MIP process heavily relies on individual annotators' interpretations, making the annotation process inherently subjective. For example, determining whether the word *discard* is metaphorical depends on how annotators perceive the concreteness or abstractness of the term it modifies. This subjectivity introduces variability in the annotations, potentially affecting the reliability and consistency of metaphor identification, especially in large-scale corpus annotation projects.

Conventional vs. Novel Metaphors One significant challenge is distinguishing between conventional and novel metaphors. Conventional metaphors, which are deeply embedded in language and culture, often align closely with basic meanings. As a result, they can behave like literal expressions, making their metaphorical nature difficult to detect using MIP. For example, the phrase *spending time* is metaphorical, drawing on the conceptual metaphor time is money, but its conventional usage renders it almost indistinguishable from literal expressions. Novel metaphors, on the other hand, are more creative and less predictable, making them more easily identified but introducing additional complexity in interpretation.

Inter-Annotator Agreement Inconsistencies among annotators further complicate the reliability of MIP. Studies, such as those by Shutova and Teufel (2010), report moderate inter-annotator agreement scores (e.g., 0.64), indicating substantial variation in the interpretation and annotation of metaphors. These discrepancies highlight the need for more robust training, clearer annotation guidelines, and tools to support annotators in making consistent decisions.

Despite its limitations, MIP remains a cornerstone in the field of metaphor identification. Its structured, step-by-step methodology bridges linguistic theory and practical application, providing a foundation for both manual annotation and computational approaches. While challenges such as subjectivity, ambiguity, and the detection of conventional metaphors persist, ongoing advancements in NLP technologies and annotation practices offer promising solutions. By refining MIP through the integration of modern techniques such as PLMs, frequency-based metrics, and hybrid frameworks, researchers can enhance its effectiveness and expand its applicability. This evolution will ensure that MIP continues to play a vital role in understanding and processing the figurative dimensions of language, supporting advancements in both theoretical linguistics and computational metaphor research.

2.2 Computational Methods for Metaphor Processing

Linguistic metaphor processing focuses on two primary tasks: metaphor identification and metaphor interpretation (Shutova 2015). These tasks are central to computational metaphor studies and aim to uncover how metaphors are expressed and understood in text (Martin 1990, Shutova 2010, Mohler et al. 2013, Ghosh et al. 2015, Rai et al. 2019, Su, Fukumoto, Huang, Li, Wang & Chen 2020).

Metaphors can appear as individual words or phrases, or they may span multiple expressions within a broader narrative. The identification task involves determining whether a given expression is metaphorical, while the interpretation task deciphers its intended meaning. Both tasks depend heavily on the surrounding context, which influences how metaphors are recognized and understood.

Effective metaphor processing often requires integrating linguistic, cognitive, and cultural knowledge. Metaphors frequently rely on shared experiences or cultural references, and their interpretation can vary depending on the background knowledge of the writer and reader. This complexity underscores the importance of context and common ground in understanding metaphorical language.

2.2.1 Metaphor Identification

Metaphor identification is a crucial task in natural language processing (NLP) that involves determining whether a given expression conveys metaphorical or literal meaning. Research in this area has evolved through various approaches, categorized into sentence-level, relation-level, and token-level tasks, each addressing different granularities of metaphorical language.

Sentence-Level Metaphor Identification

Sentence-level metaphor identification examines whether an entire sentence contains metaphorical content, and is typically framed as a binary sentence classification task (metaphorical vs. non-metaphorical). Early studies, such as those by Krishnakumaran & Zhu (2007), relied heavily on lexical resources like WordNet (Miller 1995) and statistical measures such as bigram counts from large corpora (Brants 2006). While effective for simple noun and adjective-based metaphors, these methods struggled with polysemous words and complex contextual dependencies.

Later works introduced cross-lingual methods, exemplified by Tsvetkov et al. (2014), who proposed a semantic feature-based classifier to detect metaphors in languages with limited resources. By leveraging abstractness degrees, semantic categories, and named entity types, this approach achieved F1 scores of 0.76 and 0.78 on Russian and English datasets, respectively. Similarly, Mohler et al. (2013) developed domain-specific classifiers using semantic signatures from WordNet and Wikipedia, demonstrating the potential for clustering domain-specific metaphorical patterns.

Recent advances have integrated sensory features, such as modality norms, into machine learning models. Modality norms are psycholinguistic ratings that indicate the degree to which words evoke sensory experiences (e.g., vision, hearing, touch, taste, smell, interoception, or action). For instance, the word *bright* is strongly associated with the visual modality, whereas *loud* is associated with the auditory modality. Wan et al. (2020) incorporated these modality norms into distributional representations by concatenating them with GloVe embeddings (Pennington et al. 2014). In practice, each word vector was augmented with a fixed-length vector of modality strength ratings, thereby enriching purely statistical embeddings with embodied, perceptual information. When applied to metaphor detection, this hybrid representation outperformed BERT-based baselines trained directly on contextual embeddings. Crucially, the performance gains stemmed from modeling shifts in sensory modalities between source and target concepts in metaphorical expressions. For example, *sweet smile* uses the taste modality to describe the vision modality. Detecting certain concept transfer provides a signal of metaphorical usage (Wan & Xing 2020).

Relation-Level Metaphor Identification

Relation-level approaches focus on metaphorical patterns in word pairs, such as subject-verb, verb-object, or adjective-noun relationships. Tsvetkov et al. (2014) demonstrated that lexical semantic features, including abstractness, imageability, and supersenses, could reliably detect metaphors using a random forest classifier. Shutova et al. (2016) extended this idea by incorporating visual embeddings derived from convolutional neural networks (CNNs), enhancing the model's ability to handle multimodal metaphorical associations.

Attribute-based semantic representations have also been explored. Bulat et al. (2017) demonstrated that these representations capture abstract properties better than traditional dense embeddings like Word2Vec (Mikolov et al. 2013). However, selecting appropriate levels of abstraction remains a challenge. Song et al. (2020) addressed nominal metaphors using knowledge graph embeddings, encoding metaphorical relationships as triplets, though this approach was limited to specific domains.

Ge et al. (2022) proposed a multi-task learning framework inspired by Conceptual Metaphor Theory (CMT), achieving state-of-the-art performance on datasets like MOH and TSV. By explicitly modeling the mappings between source and target concepts, this framework demonstrated the potential for enhancing relation-level metaphor identification.

Token-Level Metaphor Identification

Token-level metaphor identification targets individual words or phrases within a sentence. Initial methods, such as Do Dinh & Gurevych (2016), utilized feedforward neural networks (FNNs) for metaphor detection, achieving promising results on the VUA dataset. Gao et al. (2018) introduced BiLSTM architectures with contextual embeddings, significantly improving performance by capturing sequential dependencies in text.

To address the reliance on external resources, Gong et al. (2020) combined linguistic features such as part-of-speech tags, concreteness scores, and WordNet attributes with contextualized embeddings in a novel framework. Similarly, Chen et al. (2021) proposed a contextual inconsistency-based approach, calculating distributional distances between target words and their contexts to differentiate metaphorical from literal uses.

Recent work by Mao et al. (2022) integrated multi-task learning, combining metaphor identification with part-of-speech tagging. This approach achieved the highest F1 scores on the VUA-A dataset, highlighting the efficacy of auxiliary tasks in enhancing token-level metaphor detection.

Challenges and Emerging Trends

Despite above advancements, several challenges persist in metaphor identification: **Extended Metaphors and Multi-Word Expressions (MWEs)**: Detecting extended metaphors, which involve multiple source-target mappings, requires holistic textual analysis. Similarly, metaphorical MWEs present challenges for token-level models that rely on limited context (Rohanian et al. 2020). **Multilingual and Cross-Lingual Applications**: Although cross-lingual studies like Tsvetkov et al. (2013) show promise, most research focuses on English, limiting the adaptability of models to other languages. **Integration of Multimodal Features**: Visual and sensory modalities (Shutova et al. 2016, Kehat & Pustejovsky 2021) offer rich context but remain underexplored in metaphor identification frameworks.

Emerging trends include leveraging pre-trained language models (PLMs) such as BERT (Devlin et al. 2019), RoBERTa (Liu 2019), and GPT-style models for richer contextual embeddings. Multi-task learning and graph-based methods are increasingly applied to fuse diverse features and enhance generalization across tasks (Le et al. 2020, Mao & Li 2021). Future research aims to address these challenges by incorporating broader contextual analysis, extending metaphor identification to diverse languages, and refining techniques for detecting complex metaphorical constructs.

In summary, metaphor identification has evolved significantly, with advancements in computational methods and linguistic theory driving progress. The integration of multimodal and cross-lingual approaches, coupled with state-of-the-art machine learning techniques, holds promise for further breakthroughs in this dynamic field.

2.2.2 Metaphor Interpretation

Metaphor interpretation tasks can be broadly categorized into property extraction, word-level paraphrasing, and explanation pairing approaches. These methods address different aspects of interpreting metaphors, ranging from identifying shared attributes between source and target domains to generating literal explanations or paraphrases.

Property Extraction

Property extraction focuses on identifying shared attributes linking source and target concepts in a metaphor. Early models, such as Su et al. (2015, 2017), used lexical resources and semantic similarity measures to extract properties, typically representing them as adjectives. Later work integrated cultural and emotional dimensions into property extraction, such as Rai et al. (2019), who incorporated emotion analysis, and Su, Peng, Huang & Chen (2020), who employed culture-specific knowledge graphs. While effective for noun-based metaphors, these approaches often struggled with complex syntax and extending to other parts of speech.

Word-Level Paraphrasing

Word-level paraphrasing aims to convert metaphors into their literal counterparts by replacing metaphorical words with contextually appropriate literal terms. Shutova (2010) introduced the concept by using co-occurrence patterns and WordNet relations to rank paraphrases. Mao et al. (2018) expanded this with unsupervised methods and Word2Vec embeddings, later transitioning to pre-trained language models (Mao et al. 2022). These approaches improved downstream NLP tasks, such as sentiment analysis, but struggled with nuanced meanings and multi-word expressions (MWEs).

Explanation Pairing

Explanation pairing links metaphors to predefined explanations or paraphrases. Martin (1990) introduced a grammar- and knowledge-based system, while Bizzoni & Lappin (2018) used deep learning models to pair metaphorical sentences with literal paraphrases. Mao et al. (2022) advanced this by employing dependency parsing and curated dictionaries to interpret MWEs, demonstrating improved performance in idiomatic metaphor detection and sentiment analysis. However, these methods remain limited by their reliance on annotated data and predefined knowledge.

Neural Interpretability

While these approaches have laid the foundation for computational metaphor interpretation, they often rely on static lexical resources or curated knowledge bases, which limit their coverage and adaptability. With the rapid development of generative models, these extraction- and matching-based tasks have been significantly enhanced. Large language models can now be incorporated into the predefined explanation paradigm, providing richer, more context-sensitive paraphrases and explanations.

At the word level, this trend is complemented by the use of interpretability-oriented tools such as autoencoders and feature attribution methods, which can decompose a model's internal embeddings into human-interpretable components. By combining these decomposed semantic features with predefined explanation templates or categories, we can more effectively compare a target word's internal representation with its literal meaning, and more intuitively observe decomposition of the complex semantics of metaphors.

Beyond metaphor-specific methods, recent advances in neural network interpretability offer promising tools for metaphor interpretation. Sparse Autoencoders (SAEs) are a form of unsupervised neural network that reconstructs input data while enforcing a sparsity constraint on the hidden layer—ensuring that only a small subset of hidden units are active for any given input. This encourages the model to learn compact, disentangled, and human-interpretable representations. Sparse feature learning has its roots in sparse coding approaches (Olshausen & Field 1997) and has since evolved to include neural network models trained with sparsity penalties (Lee et al. 2007, Nair & Hinton 2009).

In recent work on interpretability, SAEs have been used to decompose hidden activations in large language models into interpretable features (Cunningham et al. 2023). By learning a sparse "dictionary" of latent features, each unit in the hidden layer tends to correspond to a specific semantic or syntactic pattern. This enables researchers to trace how particular neurons contribute to the model's behavior, making SAEs a valuable tool for understanding and explaining complex model decisions—such as those involved in metaphor interpretation. We build on this line of work in Chapter 5, where we apply sparse feature decomposition to investigate metaphor-specific activations in LLMs.

2.2.3 Datasets

Given the centrality of data to the development and evaluation of metaphor detection methods, this section provides a detailed description of the four major datasets used in this thesis: TroFi, VUA-18, VUA-20, and MOH-X. For each dataset we summarize its construction procedure, source material, annotation methodology, size and splits, inter-annotator agreement (where available), and its strengths and limitations. Choi et al. (2021) conducted a comprehensive statistical analysis of these datasets, reporting detailed counts of sentences, metaphorical tokens, and metaphor proportions across training, validation, and test splits. Table 2.1 summarizes these statistics, providing a quantitative overview that highlights differences in scale and metaphor density. These figures are particularly useful for understanding the class imbalance challenges faced by models trained on these datasets.

Dataset	#targets	%M	#Sent	Sent len
TroFi	3,737	43.5	37,737	28.3
VUA-18$_{tr}$	116,622	11.2	6,323	18.4
VUA-18$_{dev}$	38,628	11.6	1,550	24.9
VUA-18$_{te}$	50,175	12.4	2,694	18.6
VUA-20$_{tr}$	160,154	12.0	12,109	15
VUA-20$_{te}$	22,196	17.9	3,698	15.5
MOH-X	647	48.7	647	8

Table 2.1: *Detailed statistics on datasets used in this thesis. #targets is the number of tokens of target words, %M is the percentage of metaphorical targets, #Sent is the number of sentences, and Sent len is the average length of sentences.*

TroFi

The TroFi (Tropes and Figurative Language) dataset was introduced by Birke & Sarkar (2006) with the aim of supporting research on verb metaphor detection. It contains 3,737 sentences sampled from the Wall Street Journal corpus, covering 50 target verbs whose metaphoricity was to be determined. Annotation was performed semi-automatically: an unsupervised word sense disambiguation–based clustering algorithm first grouped usages of each verb, and the authors subsequently evaluated and refined the cluster assignments. Approximately 43.5% of all sentences were annotated as metaphorical.

TroFi remains one of the earliest large-scale resources for verb metaphor detection and has been widely used as a benchmark in early neural and non-neural approaches. However, the use of unsupervised clustering means that annotation quality may vary, and inter-annotator agreement (IAA) figures are not reported, making the dataset somewhat noisy for fine-grained analysis.

VU Amsterdam Metaphor Corpus (VUA)

The VUA corpus, introduced by Steen (2010), is the largest all-word metaphor-annotated corpus to date. Sentences were sampled from the British National Corpus (BNC) across four genres: news, academic writing, fiction, and conversation. Following the Metaphor Identification Procedure (MIP), annotators labeled each content word for metaphoricity. The version of VUA used in the 2018 shared task (VUA-18) was the first publicly released benchmark for metaphor detection, publishing 10,567 sentences, of which 11.6% targets are metaphorical language. In 2020, an enhanced version (VUA-20) was released, expanding the corpus to 15,807 sentences with 12.7% of the target words labeled as metaphorical. This extension provides a richer training resource and a slightly higher metaphor density. Importantly, the corpus also distinguishes between different metaphor types:

- Indirect metaphors, where contextual and basic meanings contrast. Example: *Professional religious education teachers likeMarjorie B Clark (Points of View, today) are doing <u>valuable</u> work in many secondary schools ...*
- Direct metaphors, which explicitly signal comparison. Example: *... he's like a <u>ferret</u>.*
- Implicit metaphors, where metaphorical meaning is inferred from discourse context. Example: *Naturally, to embark on such as step is not necessarily to succeed in realizing <u>it</u>.*
- Borderline cases, where annotators were uncertain or disagreed. Example: *But by the time I had turned off the road from Bellingham at Kielder village and driven <u>up</u> the bumpy Forest Drive to East Kielder Farm ...*

VUA is regarded as the gold standard for supervised metaphor detection due to its size, coverage across genres, and systematic use of MIP. It has been widely adopted in shared tasks and neural metaphor detection research. Its main limitation is class imbalance, with only a small proportion of tokens labeled as metaphorical, which can bias models toward literal predictions.

MOH and MOH-X

Mohammad et al. (2016) developed the MOH dataset by selecting verbs with 3–10 senses from WordNet and collecting corresponding example sentences. Each sentence was annotated for metaphoricity by 10 annotators on the CrowdFlower platform, and only sentences with $\geqslant$70% agreement were retained. The resulting dataset contains 1,639 sentences (1,230 literal, 409 metaphorical) involving 440 unique verbs. Shutova et al. (2016) subsequently derived the MOH-X dataset from MOH by extracting verb–subject and verb–object pairs, discarding pronominal or clausal cases, resulting in 647 verb–noun pairs (316 metaphorical, 331 literal).

MOH-X is smaller and more balanced than VUA, making it well-suited for evaluation and lexical-level studies. Its main limitation lies in its relatively narrow coverage, as it focusses on a subset of verbs with sufficient WordNet senses, and the sentences are relatively short, reducing the challenge posed by long-range context.

Chapter 3

Towards Robust Metaphor Detection

In this chapter, we focus on token-level metaphor detection (MD), one of the three metaphor detection tasks introduced in Chapter 2. The token-level MD task is formulated as a sequence labeling problem: given an input sentence $X = [w_0, w_1, ..., w_{n-1}]$, the goal is to predict a binary label $y \in [literal, metaphor]$ for the target word w_t which subscript t is index of the position of the target word in the sentence.

In our experimental setting, following the design of widely used benchmarks such as VUA and MOH-X, a target word is explicitly marked in the input, and the model is required to classify whether this target word is metaphorical given the sentence context.

> **Input**: *As a result of the <u>stalemate</u>, the Treasury has been forced to postpone $40billion in debt auctions this week.*
>
> **Target**: *stalemate*
>
> **Index**: 5
>
> **Output**: Metaphor(y=1)

Here the input is the full sentence, with the index of the target token to be detected. The output of model is a binary decision for this specific token. In this formulation, the model focuses exclusively on single-word metaphors and assumes that the target word is pre-identified in the input. Consequently, our approach does not address the detection of multi-word metaphorical expressions (e.g., phrasal verbs or idiomatic constructions). This restriction is consistent with the design of the VUA and MOH-X benchmarks but represents a limitation of the current work and an opportunity for future research on more comprehensive metaphor detection tasks.

This chapter explores three innovative approaches to token-level metaphor detection, each addressing distinct challenges in the field. The first section §3.1 provides an overview of recent neural approaches to metaphor detection, which serve as the baselines for our experiments. We summarize representative RNN-based models, pre-trained language model (PLM) approaches, and hybrid methods that integrate linguistic theories such as MIP and SPV. This background highlights the similarities and differences between these approaches and motivates the design choices behind our proposed methods introduced later in this chapter.

The second section §3.2 introduces RoPPT (RoBERTa with Pruning on Target-Oriented Parse Trees), a novel framework designed to enhance metaphor detection by reshaping and pruning syntactic parse trees to focus on semantically relevant neighbors of the target word. This approach effectively filters out noise while retaining crucial syntactic and semantic information, leading to improved performance across diverse linguistic contexts.

The third section §3.3 presents FrameBERT, a RoBERTa-based model that integrates FrameNet embeddings to capture deeper conceptual meanings in metaphor detection. Unlike traditional models that rely on shallow semantics, FrameBERT leverages structured external knowledge from FrameNet, enhancing both accuracy and interpretability. This section highlights the importance of incorporating conceptual structures and linguistic theory into metaphor detection, paving the way for more explainable NLP models.

The final section §3.4 introduces BasicMIP, a mechanism that explicitly models basic meanings to identify metaphorical expressions. By contrasting a word's contextual meaning with its basic meaning, BasicMIP aligns closely with the Metaphor Identification Procedure (MIP) theory. This section emphasizes the importance of linguistically grounded representations, offering a principled and interpretable framework for metaphor detection.

Together, these sections demonstrate the evolution of token-level metaphor detection techniques, from context denoising and conceptual embedding integration to explicit basic meaning modeling. Each approach contributes to a more refined understanding of how metaphors function in diverse linguistic settings, advancing the field of figurative language processing and its broader applications in natural language understanding.

3.1 Related Work on Token-Level Metaphor Detection

Recent research in metaphor detection has transitioned from hand-crafted linguistic features (Hartley & Barnden 1997, Shutova et al. 2010, Turney et al. 2011, Broadwell et al. 2013, Tsvetkov et al. 2014) to deep neural models, leveraging contextualized representations from pre-trained language models (PLMs). In this section, we review representative neural approaches against which we compare our methods in following sections. We group these works into RNN-based models, PLM-based models, and hybrid approaches combining theoretical insights such as Metaphor Identification Procedure (MIP) and Selectional Preference Violation (SPV).

RNN-based Approaches

Early neural approaches to metaphor detection employed recurrent neural networks (RNNs) or convolutional neural networks (CNNs) as encoders. Wu et al. (2018) used a BiLSTM–CNN hybrid architecture with Word2Vec embeddings, augmented by POS tags and word clustering features. **RNN_ELMo** (Gao et al. 2018) combined deep contextualized embeddings from ELMo with a BiLSTM backbone, effectively capturing sequential dependencies for metaphor detection. **RNN_MHCA** (Mao et al. 2019) extended this model by integrating MIP and SPV theories and applying multi-head contextual attention over a fixed window, enabling better modeling of metaphor-related contextual cues. While effective, these shallow architectures struggle to represent the full contextual semantics of words, particularly in long-distance dependencies.

PLM-based Approaches

The introduction of contextualized PLMs such as BERT (Devlin et al. 2019) and RoBERTa (Liu 2019) enabled substantial improvements in metaphor detection. **RoBERTa_SEQ** (Leong et al. 2020) fine-tuned RoBERTa in a token-level sequence labeling setting, achieving strong performance in the VUA 2020 shared task. Su, Fukumoto, Huang, Li, Wang & Chen (2020) proposed **DeepMet**, which augments RoBERTa with global and local context windows and POS features. These approaches benefit from PLMs' ability to encode rich semantic and syntactic information, but still encode the entire sentence indiscriminately, which can dilute metaphor-relevant signals in noisy contexts.

Hybrid and Theory-Driven Models

More recent work explicitly incorporates linguistic theory or additional structural information. **MelBERT** (Choi et al. 2021) operationalizes both MIP and SPV theories within a RoBERTa-based architecture, jointly modeling contextual and literal target embeddings to improve metaphor detection. **MrBERT** (Song et al. 2021) achieves state-of-the-art performance on verb metaphor detection by encoding verb–argument relational structures within BERT representations. **MUL_GCN** (Le et al. 2020) takes a multitask approach, applying graph convolutional networks (GCNs) to jointly model metaphor detection and word sense disambiguation, leveraging shared syntactic and semantic signals.

Motivation for Our Approach

These prior methods have explored token-level metaphor detection extensively. For instance, **RoBERTa_SEQ** and **DeepMet** demonstrated the effectiveness of leveraging pre-trained language models (PLMs), paving the way for later research — today, using BERT- or RoBERTa-based architectures has become almost standard practice for this task. **MelBERT** has had a lasting influence by operationalizing

MIP and SPV theories, and our own methods also incorporate these two theoretical frameworks.

However, as highlighted by approaches such as **MrBERT** and **MUL_GCN**, training purely on the original metaphor datasets sometimes provides insufficient signal for modeling the semantics of part of complex target words. This has led to a growing trend of incorporating external information to enrich target-word understanding — for example, our **FrameBERT** method introduces frame-semantic knowledge, while our **BasicBERT** approach improves basic sense modeling through more efficient use of training data.

Moreover, existing token-level metaphor detection approaches process the entire sentence without explicitly filtering out irrelevant context, which can introduce noise. Our proposed **RoPPT** method addresses this limitation by leveraging pruned parse trees to retain only syntactically and semantically relevant parts of the sentence, thereby reducing noise and amplifying metaphor-related signals.

3.2 Metaphor Detection with Effective Context Denoising

Metaphor detection in NLP requires understanding complex relationships between abstract concepts and their linguistic context. However, existing methods often struggle with filtering irrelevant noise while modeling relevant contextual information. This section introduces RoPPT, a novel framework that reshapes and prunes syntactic parse trees to focus on semantically relevant neighbours of the target word.

Key contributions of this section:

1. Target-Oriented Parsing for Context Denoising
    - Instead of considering full syntactic trees, RoPPT restructures parse trees to prioritize words most relevant to the target metaphorical expression.
    - This ensures that only the most meaningful context is retained, reducing the impact of irrelevant sub-sentences and extraneous linguistic elements.

2. A Robust, RoBERTa-Based Metaphor Detection Model
    - RoPPT integrates its target-oriented parse tree representations into a RoBERTa-based neural framework, achieving state-of-the-art performance on three benchmark metaphor datasets (VUA, MOH-X, TroFi).
3. Comprehensive Evaluation of Context Denoising Strategies
    - RoPPT is extensively tested against existing context denoising methods, demonstrating its superiority in retaining useful linguistic features for metaphor detection.
    - It surpasses existing denoising and pruning techniques while maintaining computational efficiency.

By introducing a target-centered, syntactically aware approach to metaphor detection, this section highlights the importance of efficient context modeling in NLP. RoPPT's framework not only enhances metaphor recognition but also lays the groundwork for future research in figurative language processing, providing a more refined understanding of how metaphors function in diverse linguistic settings.

3.2.1 Intuition and Motivation

Before diving into the computational details of our method, we first explain its underlying intuition with a concrete example. Consider the following sentence from the VUA dataset:

> `The house that when he first saw it had seemed to float on a raft of golden mist, now` <u>*`lay`*</u> `in a wilderness, amidst ragged grass and straggling bushes and trees dead from the heat.`

Here, the target word *lay* is used metaphorically, as the sentence does not literally describe a house lying down but rather conveys that the house exists in a desolate, neglected state. The main challenge is that the grammatical subject (*house*) is far from the verb (*lay*), separated by a long relative clause containing many intervening words.

Most previous approaches to token-level metaphor detection rely on encoding the entire sentence as a flat sequence when encoding contextual semantic information (e.g., using BERT or RoBERTa), which treats all tokens equally regardless of their syntactic relevance. In cases like the example above, the long-distance context introduces significant noise that may obscure the true semantic relationship between *house* and *lay*, making it difficult for the model to correctly classify *lay* as metaphorical.

Our approach, RoPPT, addresses this problem by leveraging syntactic structure. Instead of processing the full unfiltered sentence, we first construct a dependency parse tree and prune it to retain only the nodes that are syntactically relevant to the target word. This process effectively removes irrelevant tokens from the input, reducing contextual noise and strengthening the signal of the true grammatical relation (*house* $\rightarrow$ *lay*). Intuitively, this allows the model to "see" a cleaner version of the sentence where the most important cues for metaphor detection are preserved, which we hypothesize will lead to more accurate predictions.

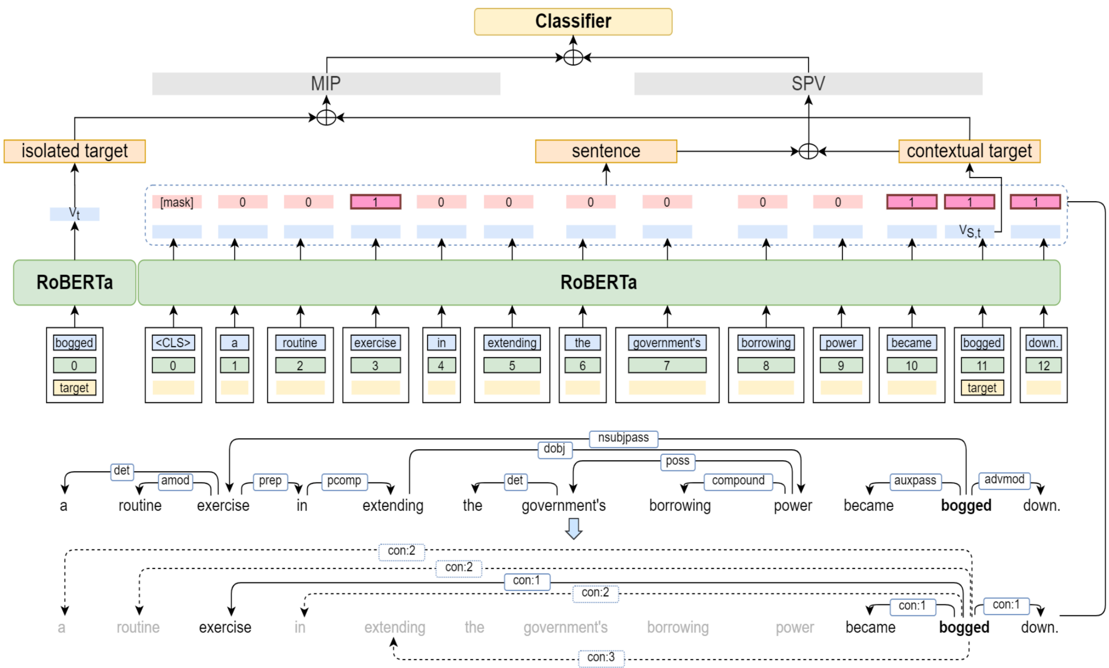


Figure 3.1: *The overall framework of RoPPT. The parse tree of a sentence is reshaped to a target-oriented tree, and the context is pruned with a pre-set threshold. The sentence embedding is the average pooling result of hidden states for pruned context from RoBERTa.* $\bigoplus$ *denotes concatenation.*

3.2.2 Method

The overall architecture of RoPPT is shown in Figure 3.1, which can be divided into two parts: a target-oriented parse tree pruning module and a RoBERTa (Liu 2019) contextual encoder.

Target-oriented Dependency Parse Tree

We tackle this challenge by introducing a target-oriented parse tree generated by three steps: **1**) reshape the original parse tree from existing parsers such as spaCy (Honnibal & Montani 2017) and Biaffine (Dozat & Manning 2016); **2**) root the tree at the target word; **3**) prune the tree according to the distance between leaves and the new root, for which we coin the term *neighbour range*. The rationale behind is that the target word is the focus of the task rather than the original root. So the re-rooting allows us to focus on the connections between target words and their relevant context. The resulting flat, target-oriented tree structure also enables simple encoding process into the model. Figure 3.1 shows an example of our reshaped tree, which retrains words with neighbour range $con = 1$ to the root 'bogged'.

RoBERTa-based Context Encoder

We employ two metaphor identification theories in our model, i.e., Metaphor Identification Procedure (Steen 2010, MIP) and Selectional Preference Violation (Wilks 1978, SPV). In MIP, a metaphor is detected when there is a contrast between target word's contextual and literal meanings, whereas in SPV a metaphorical word is identified by the semantic difference from its surrounding words.

Therefore, we model three types of semantic representations for implementing MIP and SPV, i.e., the literal meaning and the contextual meaning of a target word, and the context meaning.

Formally, given a sentence $S = (w_0, ..., w_n)$, we first employ the RoBERTa network to produce representations for each word.

$$H = \text{RoBERTa\_Enc}(\text{emb}_{\text{cls}}, \text{emb}_0, ..., \text{emb}_n) \tag{3.1}$$

Here CLS is a special token indicating the start of an input, $H = (h_{\text{cls}}, h_0, ..., h_n)$ the output hidden states, and emb_i the input embedding for word w_i. Specifically, $\text{emb}_i = \text{emb}_w + \text{emb}_{pos}$, where emb_w is the word embedding produced by the tokenizer, and emb_{pos} the position encoding.

Context denoising with the target-oriented parse tree. When modelling sentence representation, existing works directly employ the CLS embedding as a common practice (Choi et al. 2021, Song et al. 2021). In contrast, RoPPT employs the target-oriented parse tree to retain the most relevant context for a target word when computing the sentence embedding.

Specifically, our sentence embedding is computed as follows:

$$v_S = \frac{1}{n} \sum h_i, \ i \in \mathcal{C}_n \tag{3.2}$$

Here v_S is the sentence representation; $\mathcal{C}_n$ represents the n neighbour words within the neighbour range of the target-oriented parse tree, and h_i is the hidden state of w_i. In other words, we do average pooling on the most relevant context words as the sentence representation and ignore other words in the sentence.

We also design an alternative strategy by directly masking the original input sentence to the encoder according to the pruned parse tree. Specifically, we retain only the words that are preserved in the pruned tree and replace all others with the special [mask] token. We denote this intuitive solution as RoPPT with Input Mask (**RoPPT_IM**) and discuss the performance difference between these two variants in §3.2.5. Similar to Choi et al. (2021), we use the hidden state of target word w_t as the contextual target word embedding (i.e. $v_{S,t} = h_t$), and the literal target word embedding v_t is obtained by feeding the single target word w_t to the RoBERTa network.

$$v_t = \text{RoBERTa\_Enc}(\text{emb}_t) \tag{3.3}$$

We then model SPV (h_{SPV}) by concatenating the sentence embedding v_S and contextual target embedding $v_{S,t}$, and MIP (h_{MIP}) by concatenating the contextual and literal target embeddings v_t, followed by a MLP layer (i.e. $f_1(\cdot)$ and $f_2(\cdot)$), which allows the model to learn task-specific features from the two strategies. This design ensures that SPV and MIP capture complementary aspects of metaphorical usage before being fused in later layers.

$$h_{\mathrm{SPV}} = f_1([v_S, v_{S,t}]) \tag{3.4}$$

$$h_{\mathrm{MIP}} = f_2([v_{S,t}, v_t]) \tag{3.5}$$

Finally, we combine two hidden vectors h_{MIP} and h_{SPV} to compute a prediction score $\hat{y}$, and use binary cross entropy loss to train the overall framework for metaphor prediction.

$$\hat{y} = \sigma(W^\top [h_{\mathrm{MIP}}; h_{\mathrm{SPV}}] + b) \tag{3.6}$$

$$\mathcal{L} = -\sum_{i=1}^{N} [y_i \log \hat{y}_i + (1 - y_i) \log(1 - \hat{y}_i)] \tag{3.7}$$

3.2.3 Statistical Significance Testing

To rigorously evaluate the performance improvements of our proposed methods over existing baselines, we conduct statistical significance testing on the model outputs. Significance testing is crucial in metaphor detection, where performance differences between systems are often small, and claims of superiority must be substantiated by robust statistical evidence.

Historically, the two-tailed t-test has been commonly used in the metaphor detection community for this purpose. The two-tailed t-test evaluates whether the mean of the pairwise differences in performance (e.g., F1-scores across samples) between two models differs significantly from zero. However, as noted by Dietterich (1998), Rainio et al. (2024), Jurafsky & Martin (2025), the t-test is a parametric method that assumes normality of the difference distribution and independence of samples—assumptions that may not hold for classification outcomes on a fixed test set. As such, reliance on the two-tailed t-test alone has been questioned in the broader ML/NLP community.

To address this, we complement the two-tailed t-test with an additional non-parametric method: McNemar's test for pairwise comparison of classification outcomes (Rainio et al. 2024). McNemar's test focuses on discordant pairs—cases where the two models disagree—and thus directly tests whether the models have significantly different error patterns. This approach makes no assumptions about the underlying data distribution and provides more reliable confidence intervals.

By including both parametric (two-tailed t-test) and non-parametric (McNemar) tests, we ensure that our claims of significance are robust under different statistical assumptions. In all cases, we report the p-values and indicate statistical significance at the conventional threshold ($p < 0.05$). The test set sizes and evaluation metrics (Precision, Recall, F1) are explicitly stated for reproducibility.

3.2.4 Experimental Setup

Dataset. We conduct experiments on four public benchmark datasets, as detailed in §2.2.3. These include **VUA-18**, **VUA-20**, **MOH-X**, and **TroFi**, which vary in size, sentence length, and annotation format.

Methods. For comparison, we evaluate our method against several representative baselines previously introduced in §3.1: **RNN_ELMo** (Peters et al. 2018), **RoBERTa_SEQ** (Leong et al. 2020), **MrBERT** (Song et al. 2021), and **MelBERT** (Choi et al. 2021). Unlike above methods, which encode full sentences, RoPPT leverages pruned parse trees to filter out irrelevant tokens, reducing noise and enhancing metaphor-relevant signals. In addition, we design several RoBERTa-based denoising methods for ablation comparison: **RoBERTa_tree**: apply our target-oriented pruned dependency tree to the RoBERTa_SEQ encoder without using SPV and MIP strategies. **RoChunk**: a denoising method that splits sentences by commas and keep only the comma-delimited clause that contains the target before encoding, and use RoBERTa with SVP and MIP strategies. **RoWindow**: a fixed context window baseline, keep a symmetric token window around the target (window size = 4), drop everything else, and feed the windowed context to RoBERTa with the SPV and MIP stragegies. **RoPPT_IM**: an intermediate version of our approach that applies direct masking before the encoder. **RoPPT**: our full method combining pruned

Model	VUA18			VUA20		
	Prec	Rec	F1	Prec	Rec	F1
RNN_ELMo	71.6	73.6	72.6	-	-	-
RoBERTa_SEQ	80.1	74.4	77.1	75.1	67.1	70.9
MrBERT	82.7	72.5	77.2	-	-	-
MelBERT*	79.6	76.4	77.9	76.3	68.6	72.2
MelBERT	80.1	76.9	78.5	75.9	69.0	72.3
RoBERTa_tree	78.9	76.1	77.4	74.8	68.6	71.6
RoChunk	76.6	80.0	78.2	73.9	70.0	71.9
RoWindow	78.0	78.1	78.0	75.0	68.8	71.8
RoPPT_IM	73.4	74.3	73.9	67.7	66.8	67.2
RoPPT	80.0	78.2	**79.1**	75.9	70.0	**72.8**

Table 3.1: *Performance comparison on **VUA** dataset (best is in **bold**). NB: * indicates the reproduced results of MelBERT using the original source code and setting of Choi et al. (2021). RNN_ELMo and MrBERT have no results on VUA20 in their original paper. Popular denoising methods are also compared. **RoChunk** means chunk sentence by comma on RoBERTa input, **RoWindow** means denoising by a context window (size=4). **RoPPT_IM** represent masking sentence before input to transformer encoder. The solid horizontal line separates previous work (above) from our proposed models (below). The partial dotted horizontal line further separates generic denoising strategies (above) from our RoPPT-based models (below).*

parse tree guidance with SPV and MIP strategies.

Hyperparameter. We set the hyperparameter neighbour range $con = 4$ based on the validation set results. All the parser results are based on spaCy as it performs better than Biaffine empirically (see §3.2.5 for more discussion).

3.2.5 Experimental Results

Overall results. Table 3.1 shows a comparison of the performance of our models against the baseline models on VUA18 and VUA20, respectively. It is clear that our RoPPT outperforms all baselines on VUA18 and VUA20, including the state-of-the-art model MelBERT. A two-tailed t-test was conducted based on 10 paired results from RoPPT and the strongest baseline MelBERT* on both VUA-18 ($p = 0.014$) and VUA-20 ($p = 0.019$). A McNemar's test was also performed on the prediction outputs of RoPPT and the strongest baseline MelBERT* for both VUA-18 and VUA-20, yielding highly significant results with $p < 0.001$ in both cases. Due to

Models	TroFi			MOH-X		
	Prec	Rec	F1	Prec	Rec	F1
RoBERTa_SEQ	53.6	70.1	60.7	80.6	77.7	78.7
DeepMet	53.7	72.9	61.7	79.9	76.5	77.9
MrBERT	53.8	75.0	62.7	75.9	84.1	79.8
MelBERT*	53.1	73.2	61.6	78.0	79.5	78.8
MelBERT	53.4	74.1	62.0	79.3	79.7	79.2
RoBERTa_tree	50.3	77.8	61.1	76.9	83.5	79.3
RoPPT	54.2	76.2	**63.3**	77.0	83.5	**80.1**

Table 3.2: *Performance comparison on TroFi and MOH-X datasets (NB:* ***bold*** *denotes the best result).*

environment differences and random variation, we were not able to exactly match the original reported numbers; therefore, in our statistical significance tests we used MelBERT* as the baseline.

We also compared our method against several common denoising strategies. The results show that our tree-based denoising method is more effective than other popular denoising approaches such as RoChunk and RoWindow, which are sequence-based methods. We also apply our target-oriented tree to RoBERTa_SEQ, denoted as the RoBERTa_tree model. The improvement of RoBERTa_tree over RoBERTa_SEQ on two VUA datasets (i.e. 0.3% and 0.7%) further demonstrates the utility of our tree-based denoising method.

Following the setup of Choi et al. (2021), we also conducted a zero-shot transfer learning experiment shown in Table 3.2. Specifically, our model is trained on the training set of VUA20 and directly tested on the entire Trofi and MOH-X datasets. This is intended to test the generalisation power of trained models. RoPPT shows the best performance on both datasets. The results of two-tailed t-test significance test on RoPPT against MelBERT* are $p < 0.001$ on TroFi and $p = 0.021$ on MOH-X, and the McNemar's test are $p < 0.001$ on TroFi and $p = 0.042$ on MOH-H (we cannot compare with MrBERT as the code is unavailable). It can be observed that our model gives a larger margin of improvement over the baselines on TroFi (i.e., 1.3% gain over MelBERT and 0.6% over MrBERT) than MoH-X (i.e., 0.9% gain over MelBERT and 0.3% over MrBERT).

On the Use of Combined SPV and MIP Representations A key design choice in RoPPT is to jointly model SPV and MIP representations, rather than evaluating each representation in isolation. This decision is grounded in both prior empirical evidence and the specific focus of our contribution.

First, MelBERT (Choi et al. 2021), which our framework closely follows, systematically compared SPV-only, MIP-only, and combined SPV+MIP settings. Their results demonstrated that the joint modeling approach consistently outperformed either representation used individually. The rationale is intuitive: SPV is effective for metaphors that violate selectional restrictions, whereas MIP captures shifts in contextual meaning. These two theoretical perspectives are complementary—each detects cases that the other may miss—and their combination provides a more robust signal for metaphor detection.

Second, the primary novelty of RoPPT lies in the introduction of parse-tree pruning to reduce contextual noise. This pruning operation is applied at the output level of the RoBERTa encoder, thus primarily improving the quality of the contextual embeddings that feed into MIP-based representations. Consequently, additional ablation experiments that separately evaluate SPV and MIP are unlikely to yield new insights beyond what was already established in MelBERT.

Given this empirical evidence and the theoretical grounding in prior work, we opted to retain the combined representation as the default and did not conduct further ablation experiments on SPV- or MIP-only settings.

Model performance vs. Sentence length. As the averaged sentences length of TroFi (28.3 tokens) is significantly longer than that of MoH-X (8 tokens), it is worth investigating whether our model gives more performance boost on data with longer context as it is likely to be noisier. To verify this hypothesis, we evaluated the performance boost of our RoPPT against the SOTA baseline MelBERT. Table 3.4 shows the results of VUA18 with the test set split into 3 different groups based on sentence length. The results demonstrate a clear positive correlation between performance boost and sentence length.

Impact of Parsers. neumann2019scispacy As shown in Tabel 3.3, we also investigated how the choice of parsers impacts the metaphor detection performance of our model. Specifically, following and investigating the work of Neumann et al.

(2019), we tested two parsers for constructing the target-oriented dependency parse trees, namely, the CNN-based parser Biaffine and the RoBERTa-based parser spaCy. When tested on the validation set, our model achieves 78.0% with spaCy and 77.7% with Biaffine in F1 for metaphor detection, respectively. This shows that the impact of the parse choice is relatively small for our model.

Parsers	based-on	Precision	F1
spaCy	RoBERTa	95	78.0
Biaffine	CNN	90	77.7

Table 3.3: *Impact of Parsers on the validation set.*

Case Studies. RoPPT shows its strength in the following example with the target word far away from its subject, which is correctly labeled by RoPPT but incorrectly by baseline models. For the instance with metaphorical target word *bogged*, "*a routine exercise in extending the government's borrowing power to $3.1 thousand billion became bogged down.*", the target word *bogged* is separated from its subject by a long phrase, which causes baselines (including MelBERT) to fail to detect the metaphor. Thanks to the parse tree, RoPPT links *exercise* directly to the target and produces the right label. RoPPT is also able to solve false positive examples where a target verb's neighbours mislead baselines. For the example with non-metaphorical target word *bother*, "*Well, better than that, and one of the best things about it is the fact that so few people, except the French themselves, bother with it.*", *French* (which could also be understood as a language), *bother*'s neighbour, misleads other denoising methods like RoChunk and RoWindow yet our model accurately detects its real subject *people* here.

The above two examples from VUA are both classified successfully by our method. However, our method is particularly effective in reducing the noise introduced by long contexts, which is evident on the VUA dataset. In contrast, on datasets such as MOH-X, where sentence lengths are generally shorter, the advantage is less pronounced. For example, in the sentence "*He injected new life into the performance*", all methods are able to correctly identify *injected* as a metaphor.

Sent. len.	RoPPT			MelBERT*			F1 diff.	Pruning comp.	# of Sent.
	Prec	Rec	F1	Prec	Rec	F1			
<20	76.4	74.8	**75.6**	75.0	75.2	75.1	**0.5**	10.7 / 12.3	18,515
20-40	81.8	79.9	**80.8**	79.2	79.1	79.2	**1.6**	16.4 / 29.4	17,729
>40	82.3	80.0	**81.1**	78.5	76.8	77.6	**3.5**	19.5 / 53.6	7,703

Table 3.4: *RoPPT and MelBERT* performance comparison on sentences with different length range from VUA18. 'Pruning comp.' is the comparison of the average length of (pruned) / (original) sentences.*

3.3 FrameBERT: Conceptual Metaphor Detection with Frame Embedding Learning

While transformer-based models like BERT excel at capturing contextual semantics, they often miss deeper conceptual meanings essential for metaphor detection. This section presents FrameBERT, a RoBERTa-based model that integrates FrameNet embeddings to incorporate structured external knowledge.

Key contributions of this section:

1. Integration of FrameNet for Conceptual Metaphor Detection
   - FrameBERT is the first deep learning model to incorporate FrameNet embeddings for metaphor detection, leveraging external conceptual knowledge to enhance metaphor identification.
2. A More Explainable and Interpretable Metaphor Detection Model
   - By utilizing semantic frames, FrameBERT improves explainability, providing insights into how metaphorical meanings emerge from structured conceptual relationships.
3. State-of-the-Art Performance on Multiple Benchmark Datasets
   - FrameBERT outperforms or matches the best-performing models on four public metaphor detection datasets (VUA18, VUA20, MOH-X, TroFi), demonstrating the effectiveness of incorporating external knowledge.

By integrating conceptual structure and linguistic theory into metaphor detection, this section presents a significant advancement in explainable NLP models.

FrameBERT not only enhances metaphor detection accuracy but also establishes a framework for integrating external knowledge into deep learning models, paving the way for future research in metaphor processing and beyond.

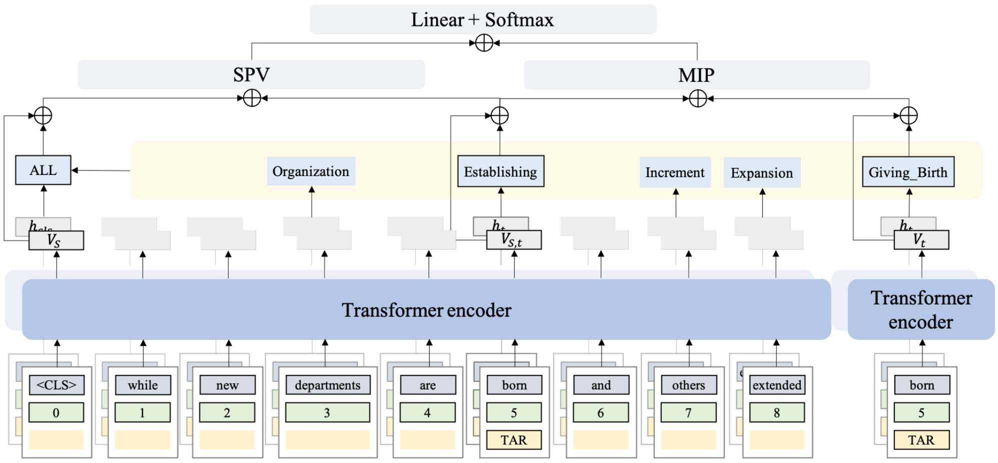


Figure 3.2: *The overall framework. The foreground encoder illustrates sentence encoder providing hidden-state representations and the background one shows concept encoder producing concepts information. The frame embedding and hidden state embedding are concatenated to make final predictions.*

3.3.1 Model

We propose FrameBERT, a novel model that can explicitly learn and incorporate FrameNet embeddings for concept-level metaphor detection. Figure 3.2 illustrates the overall architecture of FrameBERT, which consists of two components: a sentence encoder and a concept encoder.

Sentence Encoder

The sentence encoder of FrameBERT follows the architecture of RoPPT §3.2.2, as both models adopt the same theoretical framework based on the MIP and the SPV hypothesis to model metaphor detection. Specifically, both models use RoBERTa (Liu 2019) to encode the sentence representation $\mathbf{v}S$, the contextualised target word embedding $\mathbf{v}S, t$, and the isolated target word representation $\mathbf{v}_t$.

FrameBERT using MIP and SPV. Following the MIP and SPV theories, we construct vector representations $\mathbf{h}_{MIP}$ and $\mathbf{h}_{SPV}$ by concatenating the target's literal and contextual embeddings, as well as the contextual target and sentence embeddings, respectively—identical to the formulation used in RoPPT (see Eq. 3.4 & 3.5). Compared to RoPPT, FrameBERT removes the target-oriented syntactic denoising component based on dependency parse trees, streamlining the encoder. Instead, FrameBERT introduces an additional conceptual encoder to incorporate external semantic knowledge from FrameNet, which we discuss in the following subsection.

Conceptual Encoder

One of the key contributions of this method is that our model can explicitly learn and incorporate FrameNet Embeddings for concept-level metaphor detection. This is achieved via the conceptual encoder, where we first fine-tune a RoBERTa model on the FrameNet dataset (Ruppenhofer et al. 2002) with an objective to classify frame labels, and then join the conceptual encoder with the sentence encoder.

FrameNet is a rich lexical database that documents the range of semantic frames (schematic representations of situations or events) and the lexical units (words or phrases) that evoke them. Each entry in the FrameNet dataset consists of a sentence annotated with a target word (lexical unit), the semantic frame it evokes (e.g., *Commerce_buy*, *Cause_to_end*, *Coming_to_believe*), and associated frame elements (the roles related to the frame, which we do not use here). In our setting, we focus on predicting the semantic frame label associated with each target word in context. Details of the FrameNet dataset used in this work are provided in §3.3.2. The sentence, target words, and their semantic frame is shown as in Figure 3.3.

We extend the two metaphor detection theories to the conceptual dimension. For MIP we compare the literal concepts and the contextual concepts of the target word rather than its hidden-state-meaning. For SPV we compare the input sentence's concepts and target's concepts. Note that in all three cases (sentence meaning, contextual meaning, literal meaning) we allow for multiple concepts to represent the meaning; we actually use a distribution. We use FrameNet frames to provide concept information : literal concepts and contextual concepts for target words and sentence concepts for input sentence in the metaphor detection process, as shown in

Figure 3.2. Before putting the conceptual information into our metaphor detection framework, we first perform a frame identification pre-training process.

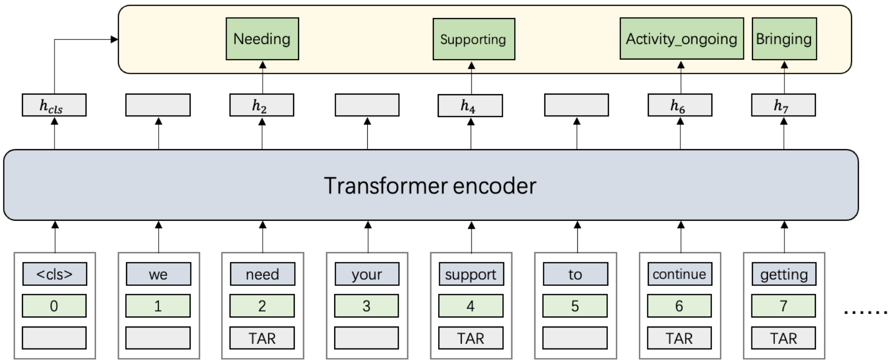


Figure 3.3: *The frame identification pre-training stage, where target word embedding are used to predict target frame and* CLS *embedding predict all frames occur in the input.*

In this pre-train stage, given an input sentence $S = (w_0, ..., w_n)$ with several targets with their frame labeled, we add a special token CLS at the beginning of the sentence and apply 12 Transformer encoder layers(like BERT) on the tokenised input to obtain the contextualised hidden states for each word $\mathbf{H} = (\mathbf{h}_{cls}, \mathbf{h}_0, ..., \mathbf{h}_n)$ and the CLS token, similar to sentence encoder part. We then leverage the contextual target word hidden states and CLS hidden states (as sentence representation) to predict the target word's frame and all frames detected in the sentence. Formally, given CLS hidden states $\mathbf{h}_{cls}$ and a list of contextualised target word hidden states $\mathbf{H} = (\mathbf{h}_0, ..., \mathbf{h}_k)$, we obtain the frame distribution for sentence and targets as follows:

$$\hat{\mathbf{y}}^{\mathbf{f}}_{cls} = \text{sigmoid}(\mathbf{W}_0\mathbf{h}_{cls} + \mathbf{b}_0) \tag{3.8}$$

$$\hat{\mathbf{y}}^{\mathbf{f}} = \text{softmax}(\mathbf{W}_1\mathbf{H} + \mathbf{b}_1) \tag{3.9}$$

where $\mathbf{W}_0$ and $\mathbf{W}_1$ are learnable parameters, $\mathbf{b}_0$ and $\mathbf{b}_1$ are bias. Note that $\hat{\mathbf{y}}^{\mathbf{f}}_{cls}$ should be applied on all frame classes, that is compute it on each possible frame class. We then minimise the following loss functions:

$$\mathcal{L}_{target} = -\sum \mathbf{y} \log \hat{\mathbf{y}}^{\mathbf{f}} \tag{3.10}$$

$$\mathcal{L}_{cls} = -\sum^{N} \sum_{i=0}^{L} \mathbf{y}_i \log \hat{\mathbf{y}}^{\mathbf{f}}_{cls} \tag{3.11}$$

$$+ (1 - \mathbf{y}_i) \log(1 - \hat{\mathbf{y}}^{\mathbf{f}}_{cls}) \tag{3.12}$$

where N is the number of training samples. L is number of frame labels, which means we are optimising the objective on all possible frame classes. We use λ as a hyperparameter controlling weights between two losses: $\mathcal{L} = \lambda * \mathcal{L}_{cls} + \mathcal{L}_{target}$; and we set it to 2 in our experiments.

After the pre-train stage, the conceptual encoder will provide frame information for metaphor detection. As shown in Figure 3.2, in the MIP module, we concatenate the contextualised frame embedding $\mathbf{h}_{S,t}$ and isolated frame embedding $\mathbf{h}_t$ of target word to $\mathbf{h}_{MIP}$ (eq. 3.13). In the SPV module, we concatenate the CLS frame embedding $\mathbf{h_{cls}}$ and contextualised target word frame embedding $\mathbf{h}_{S,t}$ to $\mathbf{h}_{SPV}$ (eq. 3.14).

$$\mathbf{h}_{MIP} = \mathbf{v}_t \oplus \mathbf{v}_{S,t} \oplus \mathbf{h}_t \oplus \mathbf{h}_{S,t} \tag{3.13}$$

$$\mathbf{h}_{SPV} = \mathbf{v}_S \oplus \mathbf{v}_{S,t} \oplus \mathbf{h}_{cls} \oplus \mathbf{h}_{S,t} \tag{3.14}$$

We then combine two hidden vectors $\mathbf{h}_{MIP}$ and $\mathbf{h}_{SPV}$ to compute a prediction score.

$$\hat{\mathbf{y}} = \sigma(\mathbf{W}_T(\mathbf{h}_{MIP} \oplus \mathbf{h}_{SPV}) + \mathbf{b}) \tag{3.15}$$

Finally, we use binary cross entropy loss to train the overall framework for metaphor prediction.

$$\mathcal{L} = -\sum_{i=0}^{N} \mathbf{y}_i \log \hat{\mathbf{y}}_i - (1 - \mathbf{y}_i) \log(1 - \hat{\mathbf{y}}_i) \tag{3.16}$$

3.3.2 Experiments

Dataset. We conduct experiments on four public benchmark datasets, as detailed in §2.2.3. These include **VUA-18**, **VUA-20**, **MOH-X**, and **TroFi**. Additionally, the concept encoder was pre-trained on **FrameNet release 1.7** (Ruppenhofer et al. 2002) with about 19k, 6k, 2k annotations for training, testing and evaluation respectively.

Baselines. We adopt several strong baselines previously introduced in §3.1, including both conventional and state-of-the-art metaphor detection models. **RNN_ELMo** (Gao et al. 2018) combined ELMo and BiLSTM as a backbone model. **RNN_MHCA** (Mao et al. 2019): they introduced MIP and SPV into RNN_ELMo and capture the contextual feature within window size by multi-head attention. **MUL_GCN** (Le et al. 2020) apply a GCN based multi-task framework by modelling metaphor detection and word sense disambiguation. **RoBERTa_SEQ** (Leong et al. 2020) is a fine-tuned RoBERTa model in sequence labeling setting for metaphor detection. **MelBERT** (Choi et al. 2021) realize MIP and SPV theories via a RoBERTa based model. **MrBERT** (Song et al. 2021) is the recent sota on verb metaphor detection based on BERT with verb relation encoded.

Models	VUA18			VUA20		
	Prec	Rec	F1	Prec	Rec	F1
RNN_ELMo	71.6	73.6	72.6	-	-	-
RoBERTa_SEQ	80.1	74.4	77.1	75.1	67.1	70.9
MelBERT ⋆	79.6	76.4	77.9	76.4	68.6	72.3
MelBERT	80.1	76.9	78.5	75.9	69.0	72.3
MrBERT	82.7	72.5	77.2	-	-	-
FrameBERT	82.7	75.3	**78.8***	79.1	67.7	**73.0***

Table 3.5: *Performance comparison on VUA datasets (best results in* ***bold****). NB: ⋆ indicates the reproduced results of MelBERT using the original source code and setting of Choi et al. (2021). * denotes our model outperforms the competing model with* $p < 0.05$ *for both 2-sided t-test and McNemar's test; except MrBERT whose code is not published.*

Ablation Study. We conducted three controlled experiments to rigorously evaluate the contribution of conceptual information in our framework. All experiments were performed on the VUA-20 dataset, using the same data splits and hyperparameter configurations as in the main experiments, to ensure comparability. The only difference lies in the treatment of conceptual embeddings or the frame identification pretraining stage.

- **Shuffling Conceptual Embeddings during Evaluation.** In this setting, we first train the model using the original conceptual encoder and its pre-trained frame embeddings, following the procedure described in §3.3.1. During evaluation, however, we randomly shuffle the conceptual embeddings within each mini-batch. This means that for every target token w_t in the evaluation batch, instead of using its true conceptual embedding $\mathbf{h}_t$, we assign the conceptual embedding of a randomly sampled target token from the same batch. This breaks the semantic alignment between the sentence context and the corresponding concept representation, while leaving the rest of the model intact. The goal of this experiment is to measure how much the model's predictive performance relies on conceptually accurate embeddings, as opposed to simply using the sentence encoder.

- **Shuffling Conceptual Embeddings during Training and Evaluation.** In this more aggressive ablation, we apply the same shuffling procedure described above not only at evaluation time, but also during training. This forces the model to learn with inconsistent conceptual information, effectively preventing it from exploiting the conceptual encoder as a reliable signal. We expect this setting to largely reduce the model to a base RoBERTa encoder with additional noise injected from the conceptual stream. This experiment tests whether the model can still derive useful information when concept supervision is corrupted throughout the entire learning process.

- **Removing Frame Fine-Tuning.** Finally, to test the contribution of the frame identification pretraining stage, we remove the frame fine-tuning step entirely and initialize the conceptual encoder with the base RoBERTa weights. The rest of the metaphor detection architecture remains unchanged. This allows us to examine whether explicitly adapting the encoder to the FrameNet frame classification task improves the downstream performance or whether the benefit is negligible.

3.3.3 Experimental Results

Overall results. Table 3.5 shows a comparison of the performance of our model against the baseline models on VUA18 and VUA20 respectively. Our model outperforms all the baseline models on VUA-20, including the state-of-the-art-model MelBERT (with $p < 0.05$ for both two-tailed t-test and McNemar's test). For VUA-18, we outperformed all the baselines significantly including the *re-produced* results for MelBERT. Table 3.7 shows the results on the MOH-X and TroFi dataset. The results show our method beats the SOTA method on the TroFi benchmark and gains comparable performance on the MOH-X dataset. In addition, we evaluated our conceptual encoder on FrameNet 1.7, achieving a frame identification Accuracy of 90.1, which still lags behind the SOTA result reported by KGFI (Su, Li, Li, Pan, Zhang, Chai & Han 2021) at the time.

Models	Accuracy
Peng et al. (2018)	89.1
Jiang & Riloff (2021)	92.1
KGFI (2021)	**92.4**
our conceptual encoder	90.1

Table 3.6: *Accuracy comparison for Frame Identification on FrameNet release 1.7 datasets (best results in **bold**).*

Ablation Study. The results of all three experiments are summarized in Table 3.8. The first experiment shows a significant performance drop (F1 decreases by approximately 13%) when conceptual embeddings are shuffled only at evaluation time, confirming that the model actively uses conceptual information during inference. In the second experiment, where shuffling occurs during both training and evaluation, the F1 score drops by a smaller margin (3.7%), suggesting that the model learns to ignore unreliable conceptual information and relies more on the sentence encoder, effectively collapsing to a baseline model. Finally, removing frame fine-tuning reduces F1 by 1.2%, provides some evidence that the frame classification pretraining may contribute to the quality of the conceptual embeddings.

Together, these results validate that **(i)** conceptual information is deeply integrated into the model's decision-making process, **(ii)** the model can fall back to purely contextual representations when concept supervision is corrupted, and **(iii)** frame-level fine-tuning improves but is not solely responsible for the model's ability to exploit conceptual signals.

	Models	Prec	Rec	F1
TroFi	RNN_MHCA	68.6	76.8	72.4
TroFi	MUL_GCN	73.1	73.6	73.2
TroFi	MrBERT	73.9	72.1	72.9
	FrameBERT	70.7	78.2	**74.2**
MOH-X	RNN_MHCA	77.5	83.1	80.0
MOH-X	MUL_GCN	79.7	80.5	79.6
MOH-X	MrBERT	84.1	85.6	**84.2**
	FrameBERT	83.2	84.4	83.8

Table 3.7: *Performance comparison of our method with baselines on TroFi and MOH-X (best results in **bold**). We do not perform a significance test since the code of MrBERT is not published.*

Concept Analysis. In this section, we illustrate how the proposed approach detects metaphors in an interpretable way and how well the method using frame features. We performed an exploratory analysis on 200 examples where our system had a correct classification, but MelBERT failed. We also identified counterexamples where MelBERT succeeded but our system did not, highlighting some of the limitations of our model. The following two examples show how frame information works in the metaphor detection procedure. The first is a true positive example with the target word in bold: '*Local people mutter and march, make speeches and throw things; staff **face** sarcasm in nearby pubs ....*' Here our system had the following concepts as the literal meaning for '*face*': '*Body_parts, Facial_expression, Change_posture*', which are more basic meanings, relating to the face as a part of the body. In contrast, contextual concepts are extracted as follows: '*Confronting_problem, Resolve_problem, Surviving*'. These capture well the contextual meaning of 'face' in the sentence. The contextual meanings are more abstract, and the contrast between literal and contextual concepts helps the model to detect the metaphorical usage of *face* here. An example of a true negative is: '*... **hot** computers are slow, the warmth might*

Models	Prec	Rec	F1
FrameBert	82.7	75.3	**78.8**
rand_in_eval	81.8	58.7	68.3
rand_in_train_&_eval	79.3	72.6	75.8
w/o frame fine-tuning	79.1	76.3	77.6

Table 3.8: *Results of ablation study, tested on VUA18. rand_in_eval represents the first experiment where the shuffle process is only performed in evaluation stage and rand_in_train_&_eval represents the second experiment where the shuffle process is performed in both training and evaluation stages. In w/o frame fine-tuning experiment, we remove the frame fine-tuning process.*

damage...'. 'Hot' is a word that can often be used metaphorically (e.g. hot topic, hot pants, hot properties), but in this sentence our model correctly identified it as literal and contextual concepts identified were identical: '*Temperature, Fire_Burning*'. In terms of how well our method using frame features, we measured the accuracy of the frame prediction module manually for these 200 examples, and found the correct frame label was identified in the top 3 frame label prediction for 178 of 200 examples (89%). This indicates our method is effective extracting frame features.

However, our method is not without limitations. In one false positive case (where our model predicted metaphorical but the gold label is literal): *Everything is home-grown, home-made and fresh: it **tastes** like forgotten food.*' Here, our model predicted metaphorical usage for *tastes*', associating it with contextual frames such as: *Experiencer_focus, Judgement_communication*', instead of more literal frames like: *Ingestion, Perception_experience*'. The confusion likely stems from the abstract phrase '*forgotten food*', which resembles metaphorical expression but is not annotated as such in the gold data. This suggests the model can be overly sensitive to figurative cues in context, leading to false positives.

3.4 Metaphor Detection via Explicit Basic Meanings Modelling

Metaphor detection relies on distinguishing between a word's contextual meaning and its more basic or concrete sense. However, we acknowledge that many words do not have a single, universally agreed-upon fundamental meaning. Due to phenomena such as homography (e.g., “bat” as an animal vs. a piece of sports equipment) and polysemy (e.g., “bridge” as a structure over water, part of a ship, or part of the nose), a surface form may correspond to multiple literal senses, which are not necessarily metaphorically related to one another. Rather than assuming a unique basic meaning per word, our approach leverages the VUA dataset, which is constructed based on MIP and provides annotated literal instances that implicitly reflect plausible basic meanings in context. BasicMIP thus constructs basic meaning representations empirically from these literal usage examples, allowing the model to learn from contextualized but non-metaphorical senses, even when ambiguity exists. This strategy avoids strong assumptions about lexical semantics while aligning with the empirical foundations of the dataset.

Key contributions of this section:

1. Explicit Basic Meaning Modeling for Metaphor Detection
   - Instead of approximating basic meanings with aggregated embeddings, BasicMIP constructs direct basic meaning representations from annotated literal instances, aligning closely with MIP theory.
2. A New Transformer-Based Model: BasicBERT
   - BasicBERT integrates BasicMIP with Aggregated MIP (AMIP) and Selectional Preference Violation (SPV), enhancing metaphor detection performance with a multi-perspective approach.
3. State-of-the-Art Performance in Metaphor Detection
   - Extensive experiments on VUA18 and VUA20 benchmark datasets show that BasicBERT outperforms existing models such as MelBERT, achieving a 1.0% F1-score improvement.
   - On VUA18, BasicBERT reaches the theoretical upper bound for targets with literal annotations, demonstrating the effectiveness of explicitly modeling basic meanings.

By redefining metaphor detection through basic meaning modeling, this section highlights the importance of linguistically grounded representations in NLP. BasicMIP provides a principled and interpretable framework for metaphor identification, paving the way for future research in metaphor processing and its broader applications in natural language understanding.

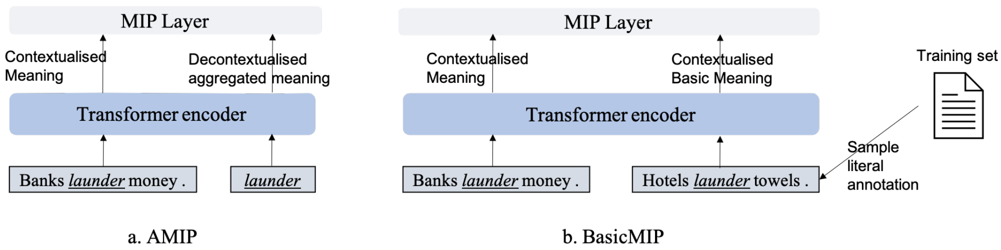


Figure 3.4: *Comparison of the AMIP implementation in Mao et al. (2019), Choi et al. (2021) and our BasicMIP.*

3.4.1 Method

BasicBERT model consists of three main components: BasicMIP, AMIP, and SPV. We include both AMIP and BasicMIP as some words do not have literal annotations in training set, so AMIP is a useful augmented component for these cases.

BasicMIP

BasicMIP, as shown in Figure 3.4, is based on MIP, in which a target word's contextualised meaning in the current context is compared with its more basic meaning. **First**, following RoPPT, we use RoBERTa to encode the input sentence and extract the contextual embedding of the target word w_t as $v_{S,t} = h_t$ (see Eq. 3.1). This step is identical to RoPPT's sentence encoding process.

Second, to contrast the contextual meaning with the basic meaning, our model learns the basic meaning representation of the target from the training annotations. According to MIP (Steen et al. 2010), we consider targets with `literal` label to represent their basic meaning. Therefore, we sample `literal` examples of the target w_t from the training set denoted as $S_b = (..., w_t, ...) \in \mathcal{U}$, where $\mathcal{U}$ is training set and S_b stands for the context sentence containing a basic usage of

w_t. Our model obtains the basic meaning embedding of w_t by feeding S_b to a RoBERTa encoder similar to Equation 3.1 and getting the t-th output hidden state h_t. The final *contextualised* basic representation of w_t is averaged among multiple literal instances, and is formulated as $v_{B,t}$, which is intrinsically different from the aggregated representations of frequent meanings typically used in prior works such as MelBERT (Choi et al. 2021) and MrBERT (Song et al. 2021). These models rely on contextualized embeddings directly obtained from pre-trained language models like RoBERTa, which are biased toward dominant or frequent word senses in the training corpus. In contrast, our approach explicitly models the basic, literal meanings by extracting representations from literal usages, providing a clearer semantic contrast for metaphor detection.

At last, we compute a hidden vector h_{BMIP} for BasicMIP, by concatenating $v_{S,t}$ and $v_{B,t}$.

$$h_{\text{BMIP}} = f_0([v_{S,t}, v_{B,t}]) \tag{3.17}$$

where $f_0(\cdot)$ denotes a linear layer to learn semantic difference between $v_{S,t}$ and $v_{B,t}$.

AMIP and SPV

The AMIP implementation of MIP theory is inherited by our model, where contextual meaning and aggregated meaning of the target are compared. Here the contextual target meaning embedding of w_t is $v_{S,t}$, the same as in Equation 3.17. Then, we feed the single target word w_t to the RoBERTa network to derive the decontextualised vector representing the aggregated meanings of w_t (Choi et al. 2021): $v_{F,t} = \text{RoBERTa}(\text{emb}_t)$.

The SPV theory is also employed which measures the incongruity between the contextual meaning of the target and its context. Similarly, the contextual target meaning embedding is $v_{S,t}$, and the context sentence meaning is produced by the CLS embedding denoted as v_S, where $v_S = h_{cls}$.

Finally, we compute AMIP (h_{AMIP}) from the contextual and aggregated target embedding, and SPV (h_{SPV}) from the contextual target meaning embedding and the sentence embedding.

$$h_{\text{SPV}} = f_1([v_S, v_{S,t}]) \tag{3.18}$$

$$h_{\text{AMIP}} = f_2([v_{S,t}, v_{F,t}]) \tag{3.19}$$

where $f_1(\cdot)$ and $f_2(\cdot)$ denote a linear layer to learn the contrast between two features.

Prediction

Finally, we combine three hidden vectors h_{AMIP}, h_{SPV} and h_{BMIP} to compute a prediction score $\hat{y}$, and use binary cross entropy loss to train the overall framework for metaphor prediction.

$$\hat{y} = \sigma(W^\top [h_{\text{BMIP}}; h_{\text{AMIP}}; h_{\text{SPV}}] + b) \tag{3.20}$$

$$\mathcal{L} = -\sum_{i=1}^{N} [y_i \log \hat{y}_i + (1 - y_i) \log(1 - \hat{y}_i)] \tag{3.21}$$

3.4.2 Experiments

Dataset. We conduct experiments on two public bench datasets: **VUA18** (Leong et al. 2018) and **VUA20** (Leong et al. 2020), which are the most popular metaphor detection benchmarks, as detailed in §2.2.3.

Baselines. We adopt several strong baselines previously introduced in §3.1, including both conventional and state-of-the-art metaphor detection models. **RNN_ELMo** (Gao et al. 2018) combined ELMo and BiLSTM as a backbone model. **RNN_MHCA** (Mao et al. 2019) introduced MIP and SPV into RNN_ELMo and capture the contextual feature within window size by multi-head attention. **RoBERTa_SEQ** (Leong et al. 2020) is a fine-tuned RoBERTa model in the sequence labeling setting for metaphor detection. **MelBERT** (Choi et al. 2021) realize MIP and SPV theories via a RoBERTa based model. **MrBERT** (Song et al. 2021) is the SOTA on verb metaphor detection based on BERT with verb relation encoded. **FrameBERT**, our proposed model introduced in §3.3, leverages frame representations from FrameNet for metaphor detection.

Hardware	TITAN RTX
Runtime/epoch	50 min
Parameters	252,839,426

Table 3.9: *Experiment details*

Models	VUA18			VUA20		
	Prec	Rec	F1	Prec	Rec	F1
RNN_ELMo	71.6	73.6	72.6	-	-	-
RNN_MHCA	73.0	75.7	74.3	-	-	-
RoBERTa_SEQ	80.1	74.4	77.1	75.1	67.1	70.9
MrBERT	82.7	72.5	77.2	-	-	-
MelBERT	80.1	76.9	78.5	75.9	69.0	72.3
FrameBERT	82.7	75.3	78.8	79.1	67.7	73.0
BasicBERT	79.5	78.5	**79.0***	73.3	73.2	**73.3***
w/o BasicMIP	81.7	75.1	78.3	74.8	69.8	72.2

Table 3.10: *Performance comparison on VUA datasets (best results in* ***bold****). NB: * denotes our model outperforms the competing model, FrameBERT, with* $p < 0.05$ *for both two-tailed t-test and McNemar's test.*

Implementation details. For target words which have no `literal` annotations in the training set, we return the decontextualised target representation as the basic meaning vector in the BasicMIP module. Therefore, the BasicMIP, in this situation, will degenerate to the AMIP implementation. The key details about the computational setup and model characteristics used in the experiment are shown in Table 3.9.

3.4.3 Results and Analysis

Overall results. Table 3.10 shows a comparison of the performance of our model against the baseline models on VUA18 and VUA20. BasicBERT outperforms all baselines on both VUA18 and VUA20, including the SOTA model MelBERT by 0.5% and 1.0% in F1 score, respectively. BasicBERT significantly outperforms MelBERT on VUA-18 ($p = 0.022$) and VUA-20 ($p = 0.006$) with two-tailed t-test; McNemar's test confirms $p < 0.001$ for both.

Ablation test. We also perform an ablation experiment to test the benefit of

	Models	Annotation	#sample	#target	F1	Acc
VUA20	w/ BMIP	has literal	18060	4076	**74.7**	91.2
		no literal	4136	2539	68.2	86.9
	w/o BMIP	has literal	18060	4076	73.3	91.0
		no literal	4136	2539	68.2	87.6
VUA18	w/ BMIP	has literal	38825	3874	**81.1**	94.7
		no literal	5122	2915	67.3	87.4
	w/o BMIP	has literal	38825	3874	80.7	94.8
		no literal	5122	2915	66.5	88.0

Table 3.11: *Breakdown results of BasicMIP.* ***has literal*** *indicates targets have* `literal` *annotations in the training set, and* ***no literal*** *indicates they have not.*

Modules	Metaphor	Literal
Contextual vs. Frequent	0.516	0.642
Contextual vs. Basic	**-0.082**	**0.809**

Table 3.12: *Contrast of features within AMIP and BasicMIP. The experiment is conducted on VUA20.*

the basic modelling. As shown in Table 3.10, the performance of BasicBERT drops substantially when removing basic meaning modelling (w/o BasicMIP) by 0.7% on VUA18 and 1.1% on VUA20, respectively.

Target with and without basic annotation Some target words in the test set might not have `literal` annotations in the training set. To better understand the mechanism of basic meaning modelling, we test the performance of BasicBERT on targets that *have* and *have not* basic meaning annotations in the training data. As shown in Table 3.11, there are 13% of samples in the VUA18 test set for which we cannot find a corresponding basic meaning annotation from the training set. This number increases to 22% for VUA20. We find BasicBERT gains significant improvement on targets with `literal` annotations from VUA20 via basic meaning modelling by 1.4% in F1 score. For these targets with `literal` annotations in the VUA18 benchmark, BasicBERT gives 81.1% in F1 score, which reaches the theoretical upper bound since the Inter-annotator agreement (IAA) value of VUA18 is around 0.8 (Leong et al. 2018) (which means further improvement might lead to overfitting).

Contrast measuring. To better compare our BasicMIP with AMIP, we conduct an experiment to directly measure the contrast between features within BasicMIP and AMIP, i.e., the contrast between the contextual and the basic meaning for BasicMIP, and the contrast between the contextual and the most frequent meaning for AMIP. Intuitively, we expect the contrast to be obvious for metaphor cases and to be slight for literal cases. Cosine distance is used to compute the contrast between two features. The contrast will fall into $(-1, 1)$, smaller numbers meaning more contrasting, larger numbers meaning less contrasting.

The results (see Table 3.12) show that the contrast of BasicMIP features is much more obvious for metaphorical samples, and there is less contrast for literal samples compared with AMIP. Moreover, AMIP only shows a minor gap of 0.13 contrast between metaphor and literal cases. However, a significant gap of 0.89 is captured by BasicMIP between metaphor and literal cases, which demonstrates that BasicMIP learns the difference between metaphorical and literal expressions well. In summary, the results show the effectiveness of basic meaning modelling in metaphor detection.

3.4.4 Case Study

We perform an exploratory analysis on metaphors where BasicMIP succeeds to detect but fails without it. Prior methods might find very simple targets difficult to classify, such as *see, back, hot*. This is mainly because their metaphorical meanings are more frequent than their basic meanings, which leads the aggregated representations dominate by metaphorical semantics. For example, *see* means *look* basically. But, *I see why you are angry* and *this place has seen the war* are even more frequent in language corpus. Therefore, the contrast with contextual meaning tends not to indicate metaphors anymore. On the contrary, basic meaning modelling learns their basic representation by focusing literal annotations directly, which enables BasicMIP to tackle them with high accuracy.

Table 3.13 shows cases where previous methods, such as MelBERT and Frame-BERT, fail but ours succeeds. Corresponding sentences with basic usage of target from the training set are also included. In addition, Figure 3.5 and Figure 3.6 provide word sense illustrations constructed using RoBERTa embeddings and visualized with

Target	Cases	Basic Examples
get	we will , i 'm just saying we do wan na get into cocktail	where do you get your carrots from ?
	they 're watching neighbours come on , get up you lazy bugger !	and you 'll get a separate room
	oh we did n't get much further on there , what we started with this morning.	i 'm gon na get some cleaning , i 'll get some cleaning fluid this week .
back	why ca n't they take it through the back door and up the stair ?	within 10 minutes i had turned my back on the corduroy battalions of trees and was striding under a still.
	they are unlikely to find a place to do so which is not in somebody 's back yard .	on the edge of the lawn with his back to the cedar tree .

Table 3.13: *Cases study of targets "get" and "back". The cases are taken from VUA20.*

principal component analysis (PCA). The sense-level annotations used to generate these figures are derived from the SemCor corpus (Miller et al. 1994), a widely used, manually sense-annotated corpus based on WordNet (Miller 1995). In the figures, each dot represents a single occurrence of the word of interest (e.g., *back* or *get*) in SemCor. The RoBERTa embedding of that occurrence was projected into 2D space using PCA. We can see the most frequent meaning of *back* is *'former location'* and *'travel backward'* instead of the basic meaning *'human body'*. And the meanings of *get* are almost equally frequent.

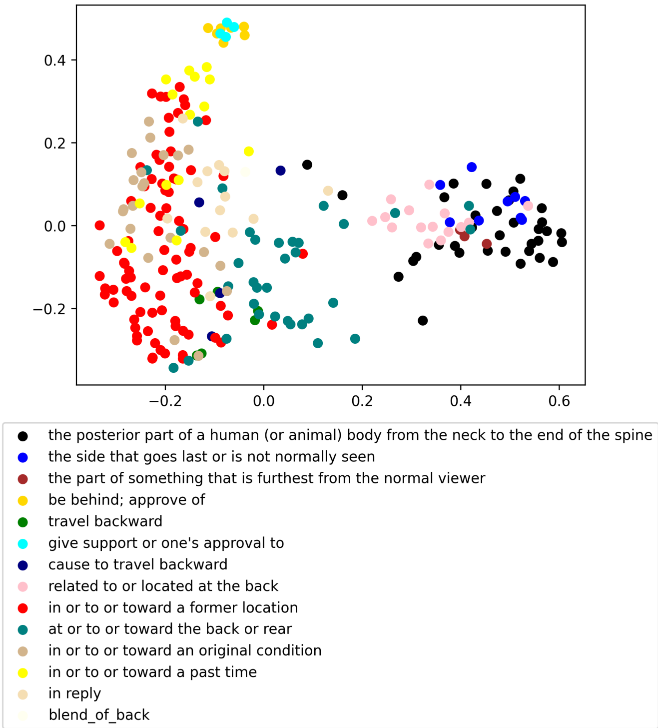


Figure 3.5: *Senses of back from word sense disambiguation dataset SemCor.*

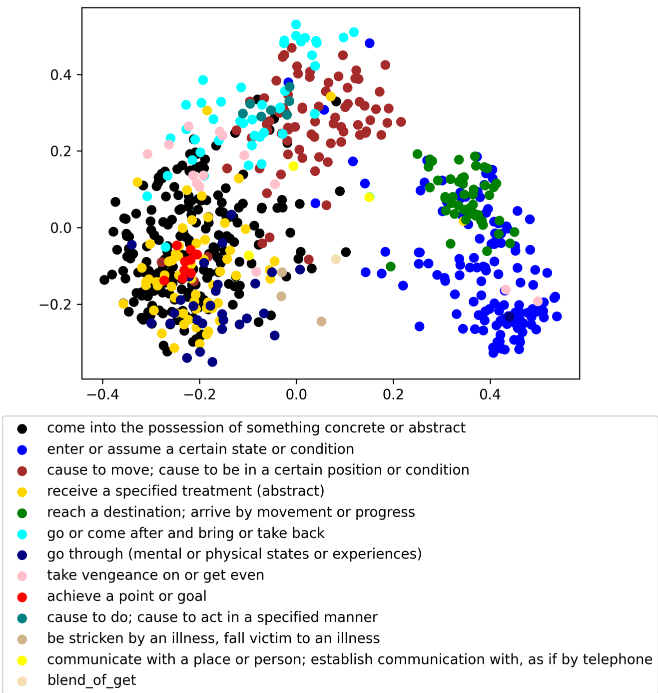


Figure 3.6: *Senses of get from word sense disambiguation dataset SemCor.*

3.5 Conclusion

This chapter presents three innovative approaches to metaphor detection, each addressing distinct challenges and advancing the field of natural language processing (NLP).

First, RoPPT introduces a novel framework for metaphor detection by reshaping and pruning syntactic parse trees to focus on semantically relevant contextual information. This target-oriented approach effectively denoises long and complex sentences, achieving state-of-the-art performance and demonstrating the importance of efficient context modeling in metaphor identification.

Second, FrameBERT leverages FrameNet embeddings to incorporate structured external knowledge, enabling concept-level metaphor detection. By integrating linguistic theory and deep learning, FrameBERT not only matches or surpasses existing models in performance but also enhances interpretability, offering a more explainable framework for understanding metaphorical expressions.

Finally, BasicBERT addresses the challenge of distinguishing between a word's contextual and basic meanings by explicitly modeling basic meanings from annotated training data. This approach aligns closely with linguistic principles and achieves state-of-the-art results, even reaching the theoretical upper bound for instances with basic annotations. BasicBERT's success highlights the value of linguistically grounded representations in metaphor detection and opens avenues for extending this approach to other forms of creative language, such as humor and sarcasm.

Discussion and Comparative Analysis

To better understand the relative strengths and weaknesses of the three proposed approaches—RoPPT, FrameBERT, and BasicMIP—we conducted a comparative analysis across the VUA-18 and VUA-20 datasets. Table 3.X reports a contingency-style summary of their performance in terms of precision, recall, and F1 score.

From these results, we observe several noteworthy patterns:

- **Precision–Recall Trade-offs**: FrameBERT achieves the highest precision on both datasets (82.7 on VUA-18, 79.1 on VUA-20), suggesting its frame-based

Dataset	Prec	Rec	F1
VUA18	80.0/82.7/79.5	78.2/75.3/78.5	79.1/78.8/79.0
VUA20	75.9/79.1/73.3	70.0/67.7/73.2	72.8/73.0/73.3

Table 3.14: *Contingency table comparing the performance of the three proposed methods. Results for RoPPT (red), FrameBERT (green), and BasicBERT (blue) are shown in distinct colors to facilitate visual comparison of their difference.*

knowledge integration reduces false positives. However, its recall lags behind, indicating a tendency to miss some metaphorical cases.

- **Balanced Performance**: RoPPT demonstrates a strong balance between precision and recall, producing the highest F1 score on VUA-18 and competitive performance on VUA-20. Its use of syntactic pruning appears to enhance recall without significantly sacrificing precision.

- **Stability Across Datasets**: BasicMIP provides a solid baseline with relatively stable performance across both datasets. Its design is closely aligned with the metaphor definitions adopted in VUA, as it explicitly models the Metaphor Identification Procedure (MIP) semantics, which allows it to capture metaphorical meaning in a manner consistent with the dataset's annotation guidelines. This explains its robustness and comparable F1 score, despite its relatively simple architecture.

This comparison shows that no single method dominates across all metrics and datasets. RoPPT's strength lies in handling long-range syntactic dependencies, FrameBERT improves precision via frame semantics, and BasicMIP remains a reliable, definition-aligned baseline.

Future Work

Given the complementary nature of these methods, a natural next step is to explore ensemble or hybrid approaches. For example, combining RoPPT's syntactic pruning with FrameBERT's frame-semantic representation may improve recall while preserving high precision. Similarly, BasicMIP can serve as a fallback classifier in cases where syntactic pruning or frame-based reasoning fails, potentially improving overall robustness. Investigating such ensemble strategies is a promising direction for future research.

Together, these contributions significantly advance the field of metaphor detection, offering robust, interpretable, and linguistically informed frameworks. We also contribute to further research in figurative language processing and its broader applications in natural language understanding.

Chapter 4

MMTE: Corpus and Metrics for Evaluating Machine Translation Quality of Metaphorical Language

Metaphors play a fundamental role in human language, enabling nuanced expression and deeper semantic understanding. However, translating metaphorical expressions across languages presents a significant challenge for machine translation (MT) systems, as direct word-to-word translations often fail to capture the intended figurative meaning. Despite advancements in MT, existing evaluation metrics primarily focus on general translation quality and fluency while largely overlooking the complexities of metaphor translation. This gap underscores the need for specialized evaluation frameworks that assess how effectively MT systems preserve metaphorical meaning across languages.

This chapter introduces MMTE (Metaphorical Machine Translation Evaluation), a novel corpus and evaluation framework designed specifically for metaphor translation assessment. MMTE provides a structured methodology for systematically evaluating the quality of metaphor translations by integrating fine-grained human and automatic evaluation metrics. Through an extensive multilingual dataset, MMTE examines how metaphors are processed in English, Chinese, and Italian, highlighting both universal conceptual patterns and culturally specific nuances.

The chapter is structured as follows. First, we outline the construction of the MMTE corpus, detailing the dataset creation, annotation guidelines, and evaluation criteria. The dataset is built using translations from multiple MT models, including commercial and open-source systems, with expert annotations ensuring high-quality assessment. Next, we introduce a fine-grained annotation framework, which evaluates translation quality along four key dimensions: Fluency, Metaphorical Equivalence, Emotion, and Authenticity. These dimensions enable a comprehensive analysis of how well metaphorical meaning is retained in translation.

Additionally, the chapter explores the impact of metaphorical equivalence on translation accuracy and emotional salience. We analyze the correlation between metaphor translation quality and emotional expression, highlighting the role of figurative language in conveying affective meaning. To further enhance evaluation, MMTE incorporates post-editing methods to generate high-quality gold-standard references, facilitating automatic evaluation metrics such as BLEU, ROUGE, and BERTScore.

Finally, we examine the performance of large language models (LLMs) like GPT-4 in metaphor evaluation, assessing their ability to classify metaphor equivalence and provide human-like annotations. The results demonstrate the increasing potential of LLMs in assisting with metaphor translation evaluation, while also revealing their limitations in handling cultural and contextual variations.

By addressing the shortcomings of existing MT evaluation metrics and introducing a specialized framework for metaphor translation, this chapter contributes to the development of more accurate, culturally adaptive, and semantically rich machine translation systems. MMTE not only provides a valuable resource for MT research but also provides a exploration for future advancements in figurative language processing across multilingual contexts.

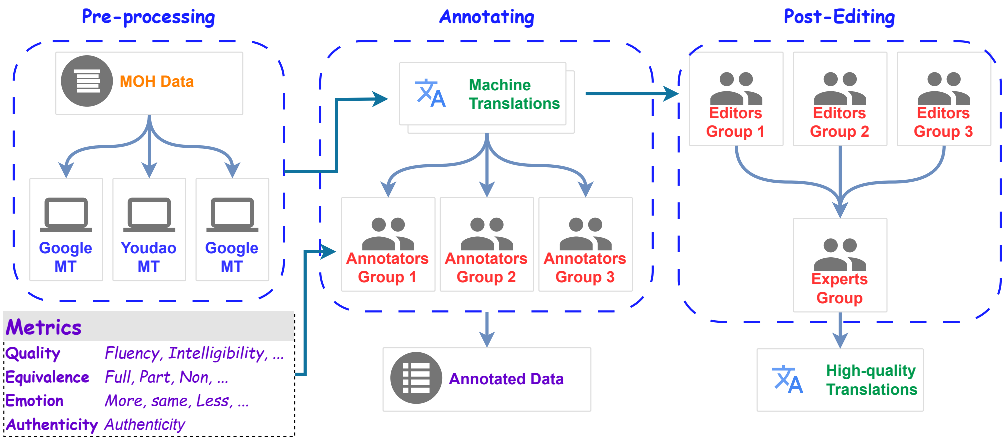


Figure 4.1: *The dataset creation framework. By translating, annotating, and post-editing, we create a cross-lingual metaphor dataset. Specific details of these sub-steps are elaborated in §4.1.1, §4.1.2, and §4.1.3, respectively.*

4.1 Metaphorical Translation Quality Annotation Framework

As discussed in §1.3.2, metaphorical expressions are not evaluated sufficiently with current MT evaluation metrics. To address this issue, we propose a set of novel MT evaluation metrics based on manual annotation and post-editing. The proposed metrics aim to provide a more accurate and insightful assessment of MT performance in handling metaphors. Our framework, as shown in Figure 4.1, allows for the evaluation of MT outputs in terms of their metaphorical expressions, enabling a more comprehensive analysis—compared to traditional machine translation evaluation methods—of how effectively nuanced metaphorical meanings are captured.

4.1.1 Initial Dataset Translation

Due to the absence of parallel multilingual metaphor datasets, we constructed our own dataset. We employ the **MOH-X** dataset (Mohammad et al. 2016) as our source, consisting of 315 metaphorical and 332 literal sentences sampled from WordNet (Miller 1998), as described in § 2.2.3. In MMTE, **Literal** samples refer to those not containing metaphors.

We utilise four popular MT models to generate translations: the **Google**

source instances	google-en-zh	youdao-en-zh	opus-mt-en-zh	GPT-4o
The scream **pierced** the night.	尖叫声**划破**黑夜。	尖叫声**划破**黑夜。	尖叫声**刺穿**了夜晚。	尖叫声**刺穿**了夜晚。
The Senator **steamrollered** the bill to defeat.	参议员**以压倒性的方式**使议案落败。	那位参议员**强行**使该法案失败。	参议员把法案**推倒**了。	参议员将该法案**压倒性地** 击败。
	google-en-it	**youdao-en-it**	**opus-mt-en-it**	**GPT-4o**
The scream **pierced** the night.	L'urlo **squarciò** la notte.	L'urlo **forò** la notte.	L'urlo **ha trafitto** la notte.	L'urlo **ha squarciato** la notte.
The Senator **steamrollered** the bill to defeat.	Il senatore ha **schiacciato** il disegno di legge per sconfiggerlo.	Il senatore ha **buttato via** il disegno di legge per sconfiggerlo.	Il senatore ha **rullato** il conto per sconfiggere.	Il senatore ha **fatto a pezzi** il disegno di legge per sconfiggerlo.

Table 4.1: *Paired samples of source instances and their machine translations from different translation models. Target verbs are in* ***bold and underlined****.*

Cloud Translation API, the **Youdao Cloud Translation API**, the open-source **Helsinki-NLP/opus-mt** model from Hugging Face, and **GPT-4o** to translate English source data into **Chinese** and **Italian**, enabling us to explore and compare the treatment of metaphors in two languages with distinct characteristics. Table 4.1 presents example metaphors paired with their translations in the two target languages. Additional information regarding preprocessing is presented in §4.2.1.

4.1.2 Metaphor Annotation Criteria

Our annotation protocol involves comparing translations with their source sentences. We hire 18 linguistics majors who are native speakers of the target languages to annotate and post-edit 647 English-Chinese (EN-ZH) and English-Italian (EN-IT) translations, with each sample being annotated by 3 individuals. Professional translators cross-checked the results, resolved disagreements in meetings, and recorded final decisions. Additional details are in §4.2.2. The source instances and their corresponding translations are systematically annotated based on four criteria to evaluate translation quality: Quality, Metaphorical Equivalence, Emotion, and Authenticity. These criteria are outlined and further broken down as follows.

Quality. To estimate the quality of the translation, we adopt criteria inspired by several existing human assessment methods for MT (Carroll 1966, Church & Hovy 1993, White et al. 1994) and consider three primary aspects of quality, including Fluency, Intelligibility, and Fidelity. Detailed definitions are presented in §4.2.3.

Equivalence. To ascertain how metaphors impact MT, we propose Equivalence to

Equivalence	Source	Target
Full	The White House **sits** on Pennsylvania Avenue.	白宫**坐落**在宾夕法尼亚大道上。
	The ex-slave **tasted** freedom shortly before she died.	l'ex schiava ha **assaporato** la libertà poco prima di morire.
Part	**Wallow** in your success!	**沉浸**在你的成功中吧！
	My personal feelings **color** my judement in this case.	i miei sentimenti personali **offuscano** il mio giudizio in questo caso.
Non	This drug will **sharpen** your vision.	这药能**改善**你的视力。
	Fire had **devoured** our home.	l'incendio **distrusse** la casa.

Table 4.2: *Instances of various Equivalence types in metaphor translation. Full refers to the same literal and contextual meanings; Part means similar contextual meanings and different literal meanings while both being metaphorical; and Non means similar contextual meanings and different literal meanings with the translation being non-metaphorical.*

describe how figurative expressions are translated into another language based on two features: 1) *How the meanings of the source and target are conveyed* 2) *Whether or not the translation is still figurative.* By comparing source texts and translations, annotators are asked to determine to what extent the target word is Equivalent in figuration. The annotators label the translation using a set of five distinct tags, encompassing three types of Equivalence and two types of Mistranslation. We elucidate the types of Equivalence in Table 4.2, based on the following definitions:

- ***Full-Equivalence***: When comparing the source and translation, both the literal meanings and the contextual meanings of the target word are the same.
- ***Part-Equivalence***: When comparing the source and translation, only the contextual meanings of the target word are similar. The literal meaning of the target word between the source and translation is different, but they are both metaphorical.
- ***Non-Equivalence***: When comparing the source and translation, only the contextual meanings of the target word are similar. However, the translation is a non-metaphorical expression, making the literal meaning of the target words between the source and translation different.

We also identify two types of mistranslation:

- ***Misunderstanding***: When the literal meaning of the target word in the source text and translation are similar, but the translation fails to convey the contextual meaning of the target word in the source language.

- ***Error***: When the target word is mistranslated, meaning that not only the contextual meanings are different between the source and translation, but their literal meanings also differ.

If the source instances are non-metaphorical expressions, annotators are instructed to only classify the translations into three categories: ***Literal***, ***Metaphorical***, and ***Error***. The non-metaphorical portion of the data is used for subsequent comparisons with the metaphorical instances.

Emotion. Inspired by Mohammad et al. (2016), we incorporate an analysis of emotion to investigate whether metaphorical expressions in translations convey additional emotional information compared to non-metaphorical expressions. By comparing a source sentence and its translation, the annotators determine to what extent the target word and its translation convey different amounts of emotion. There are four labels to judge emotion: ***Zero***, ***Less***, ***Same***, and ***More***, separately representing that the target word in the source context conveys no emotion, or that the target word in the translation conveys less, the same, or more emotion than the target word in the source sentence.

Authenticity. Authenticity is an extension of existing criterion (Doyon et al. 1999), evaluating: *To what extent the translated metaphor reads like standard, well-edited language, such that the metaphor would be understood by a native speaker of the target language.* The annotators are asked to judge this criterion on a 5-point Likert scale (Likert 1932).

4.1.3 Post-Editing

Due to the requirement for gold references by automatic evaluation algorithms like BLEU (Papineni et al. 2002) and ROUGE (Lin 2004), we introduce a post-editing method to modify the translation results of four MT models to generate a gold standard translation reference, as is common practice (Senez 1998, Allen 2003, Somers 2003). Three groups of annotators, who are native speakers of each target language, are asked to post-edit the translations, resulting in three groups of human-edited translations for both Chinese and Italian. Finally, a panel of expert translators perform final filtering to select the best quality edited translation as the gold reference. Additional details regarding the human annotation process are presented in §4.2.3. While multiple high-quality human translations were collected and are preserved in the released dataset, a single gold translation was selected specifically to serve as a reference for evaluating metaphorical equivalence. This helps us focus on capturing fine-grained distinctions in figurative meaning, since using multiple references could blur those detailed differences.

We employ several quality control methods to ensure the quality of the dataset obtained through post-editing the machine translations. Annotators compare four different translations, selecting high-quality ones or modifying low-quality ones to provide a reference translation, including translations of both metaphorical and non-metaphorical language. Three separate annotator groups work on each sample. An expert panel of translators then reviews and refines the selections. Annotators also mark the positions of target words during alignment to avoid issues in word-level processing. The final dataset includes aligned English, Chinese, and Italian translations, with 315 metaphorical and 332 literal instances per language, totalling over 1900 instances.

4.1.4 Automatic Metrics for Translation Quality

We introduce several automatic metrics to evaluate the quality of translations, which are described below.

BLEU/ROUGE. We adopt BLEU (Papineni et al. 2002) and ROUGE (Lin 2004) as standard reference-based automatic metrics to evaluate translation quality. BLEU computes n-gram precision between the candidate translation and a set of reference translations, with a brevity penalty to discourage overly short outputs. We compute the brevity penalty BP,

$$BP = \begin{cases} 1 & \text{if c>r} \\ e^{(1-r/c)} & \text{if c}\leq\text{r} \end{cases} . \tag{4.1}$$

c is the length of the candidate translation and r is the effective reference corpus length.

Then,

$$BLEU = BP \cdot \exp\left(\sum_{n=1}^{N} w_n \log p_n\right) \tag{4.2}$$

where p_n is the n-gram precision, w_n is the weight summing to 1(normally be $1/N$). We report BLEU-1 (unigram precision, measuring lexical adequacy) and BLEU-4 (up to 4-gram precision, rewarding longer sequence matches and better fluency) as shown in Table 4.7.

ROUGE, in contrast, measures recall rather than precision, making it more sensitive to coverage of reference content. We specifically use ROUGE-L, which is based on the length of the Longest Common Subsequence (LCS) between a reference text X and a system-generated text Y. Let $LCS(X, Y)$ denote the length of their longest common subsequence. The recall, precision, and F-measure are defined as follows:

$$R_{\text{LCS}} = \frac{LCS(X, Y)}{|X|} \tag{4.3}$$

$$P_{\text{LCS}} = \frac{LCS(X, Y)}{|Y|} \tag{4.4}$$

$$F_{\text{LCS}} = \frac{(1 + \beta^2)\, R_{\text{LCS}}\, P_{\text{LCS}}}{R_{\text{LCS}} + \beta^2 P_{\text{LCS}}} \tag{4.5}$$

where β is a parameter that balances the importance of recall and precision (typically $\beta = 1$).

Together, BLEU and ROUGE provide complementary views: BLEU rewards precision (avoiding spurious additions), while ROUGE rewards completeness (avoiding omissions).

BERTScore. To handle cross-lingual translation evaluation and reduce reliance on exact string matching, we employ BERTScore (Zhang et al. 2019), which computes a semantic similarity score between candidate language translation and original language reference by aligning contextual embeddings token-by-token using cosine similarity. BERTScore is particularly effective in cross-lingual settings (Song et al.

2021a) because it relies on shared representation space rather than exact word overlap.

GPT score. We also employ GPT-4o[1] as an annotator to score the translation results using the same scoring criteria as human annotators. We prompt GPT-4o with a carefully designed evaluation template (included in §4.2.4) asking it to rate translation quality on a 1–5 Likert scale.

4.2 Corpus Creation

This section provides a comprehensive overview of how we create the corpus, including MT model selection, data preparation, and implementation details. We outline the methodologies used to conduct our experiments and ensure reproducibility of the results.

4.2.1 Translators and Languages

The **Google Cloud Translation API** (Translation V3 API) is a prominent commercial multilingual translation tool employing neural MT (NMT) techniques, known for its wide-ranging capabilities and comprehensive language support.

The **Youdao Cloud Translation API** is a popular commercial multilingual NMT tool within the Chinese community, proficient in handling Chinese language translation tasks.

The **Helsinki-NLP/opus-mt** models are pre-trained on the open parallel corpus (OPUS), a continuously expanding collection of translated texts sourced from the web. These models are widely used by researchers and practitioners due to their effectiveness and versatility.

The **GPT-4o**, developed by OpenAI, is an advanced language model designed to perform a wide range of natural language processing tasks, including serving as a highly capable translator model that can handle multiple languages with high accuracy and fluency.

[1] gpt-4o-2024-05-13 https://platform.openai.com/

Chinese, a Sino-Tibetan language, is renowned for its rich idiomatic expressions and extensive use of metaphors. **Italian**, a Romance language descended from Latin and belonging to the Indo-European language family like English, provides a distinct comparison. The distinction between these target languages enables a more accurate assessment of the models' performance in preserving metaphorical meaning.

4.2.2 Annotation Setup

Our annotation platform is built on a private server using an open-source annotation tool - Doccano (Nakayama et al. 2018). We hired 18 annotators who are native speakers of the target languages, all of whom are linguistics majors with professional working competency in English. All annotation workers are paid based on the median wage of similar tasks on Amazon Turk, which is 10 dollars/hour. Specifically, the annotators are divided into six groups, each with three annotators. The groups are further equally divided between annotating English-Chinese (**EN-ZH**) instances and English-Italian (**EN-IT**) instances. Each group is tasked with labelling target words and post-editing the entire MOH-X dataset and its translations with all 647 pairs of data, resulting in each paired instance being annotated three times. In the final step, all annotation results are cross-checked by professional translators. A group meeting was held to discuss instances of disagreement, and final decisions were recorded on an online discussion website for future reference.

4.2.3 Guideline

Before starting the annotation process, we provide each annotator with a detailed annotation guideline, including examples. Additionally, we require them to annotate a small portion of extra data as a practice session before officially beginning the task. Below is a section of the guideline we provide.

> Before proceeding with the evaluation, please read the following guidelines carefully. In each annotation sample, an **English** sentence is be given as source text, followed by four translations in **Chinese**. Please evaluate each translation based on the criteria listed below. You will also be asked to supply your own

translation as the gold reference.

Sentence Quality

Please compare the source sentence and its translation without reference to the correct translation, and evaluate the translation from following aspects:

- ***Fluency***: To what extent the translation is well-formed and grammatical, ensuring that it sounds like it was originally written in the target language.
- ***Intelligibility***: To what extent the translation is easily understood and conveys metaphorical meaning sufficiently, such that readers can gain the intended interpretation.
- ***Fidelity***: The extent to which the translation is faithful to the source sentence, such that there is minimal distortion, twisting, or altering of meaning.
- **Overall**: An overall assessment to indicate the quality of the entire sentence seen as a whole.

Please judge these four aspects of quality on a 5-point Likert scale: 5) Very Good; 4) Good; 3) Acceptable; 2) Poor; 1) Very Poor.

Equivalence

Please compare the source sentence and its translation, and determine to what extent the target word and its translation are Equivalent in figuration. Here are the definitions of the three types of Equivalence and two types of mistakes:

- **Full-Equivalence**: When comparing the source sentence and translation sentence, both the literal meanings and the contextual meanings of the target word are the same.
  EN: He @injected@ new life into the performance.
  ZH: 他给表演注入了新的生命

- **Part-Equivalence**: When comparing the source sentence and translation sentence, only the contextual meanings of the target word are similar. The literal meaning of the target word between the source sentence and translation are different, but they are both metaphorical.
  EN: @Wallow@ in your success!
  ZH: @沉浸@在你的成功中吧！

- **Non-Equivalence**: When comparing the source and translation, only the contextual meanings of the target word are similar. However, the translation is a non-metaphorical expression, making the literal meaning of the target word between the source and translation different.
  EN: Sales were @climbing@ after prices were lowered.
  ZH: 价格下跌后销售额@上升@。

- **Misunderstanding** When the literal meanings are similar between the target word in the source text and the target word in the translation, but the translation conveys no contextual meaning like the target in the source language.
  EN: I @attacked@ the problem as soon as I got out of bed.
  ZH: 我一下床就@攻击@了问题

- **Error**: When the target word is mistranslated, meaning that not only the contextual meanings are different between the source and translation, but their literal meanings also differ.
  EN: @Stamp@ fruit extract the juice.
  ZH: @果果@提取果汁。

Emotion

Please compare the source sentence and its translation, and determine to what extent the target word and its translation convey equal amounts of emotion. There are four labels to judge emotion:

- **Zero**: If the target word in source context conveys no emotion, please fill **Zero**.
  EN: I can not @digest@ milk products.
  ZH: 我不能消化牛奶产品。

- **More**: The target word in the translation conveys ***more*** emotion than the target word in the source sentence.
  EN: The seamstress @ruffled@ the curtain fabric.
  ZH: 裁缝女把窗帘布弄得一团糟.

- **Same**: The target words in the two sentences convey a ***similar*** degree of emotion.
  EN: I @salute@ your courage!
  ZH: 我向你的勇气致敬!

- **Less**: The target word in the translation conveys ***less*** emotion than the target word in the source sentence.
  EN: The spaceship blazed out into space.
  ZH: 太空船飞向太空

Authenticity Target

Please compare the target in the source sentence and its translation, and evaluate whether the target translation is authentic. In other words, to what extent is the translation idiomatic (i.e. is expressed in a way that a native speaker would express it)? Please judge the target on a 5-point scale: 5) Very Good; 4) Good; 3) Acceptable; 2) Poor; 1) Very Poor.

Post-Editing

By referring to the source sentence and its translations, in addition to the above Equivalence scale, please give two fluent and high-quality translations: 1) using figurative language (full-equivalence, part-equivalence) and 2) without using figurative language (non-equivalence). You should focus on the given target word, and make sure it is translated into an appropriate expression.

4.2.4 GPT Prompt

In order to complement human evaluation and ensure scalability of our assessment process in the future work, we leverage GPT-4o as an automatic evaluator, effectively acting as a virtual annotator. The prompt shown in Table 4.3 was carefully designed to align with the evaluation criteria used by human annotators, focusing on three key dimensions: Fluency, Intelligibility, and Fidelity. These dimensions mirror the aspects considered in our manual evaluation protocol, ensuring that the automatic scores are comparable to human judgments.

GPT-4o receives the source sentence and candidate translation as input and outputs a single overall quality score on a 1–5 scale. The scoring rubric follows the same guidelines provided to human annotators, where higher scores indicate better semantic preservation and naturalness. This design enables us to directly compare GPT-4o's judgments with human ratings, providing a reproducible automatic metric that can be used for automatic translation evaluation.

Prompt	You are a professional translation evaluator. Your task is to evaluate a candidate translation with its source sentence and assign a quality score on a scale of 1–5, from following aspects: 1. Fluency : To what extent the translation is well-formed and grammatical, ensuring that it sounds like it was originally written in the target language. 2. Intelligibility : To what extent the translation is easily understood and conveys metaphorical meaning sufficiently, such that readers can gain the intended interpretation. 3. Fidelity : The extent to which the translation is faithful to the source sentence, such that there is minimal distortion, twisting, or altering of meaning. Scoring Guidelines: 5 – Very Good: meaning fully preserved, fluent and natural. 4 – Good: minor errors but meaning mostly preserved and still natural. 3 – Acceptable: some meaning lost or awkward phrasing but understandable. 2 – Poor: significant meaning loss or poor fluency. 1 – Very Poor: completely incorrect or nonsensical translation. Instructions: - Focus on mentioned three aspect: Fluency, Intelligibility, and Fidelity. - Return an overall assessment to indicate the quality of the entire sentence seen as a whole. - Do not explain your reasoning; output only the final numeric score (1–5).
Query	**Source Sentence**: {source_sentence} **Candidate Translation**: {candidate_translation}

Table 4.3: *Prompt and Query Format for LLMs in GPT-4o evaluation.*

Language	Source	Metaphorical	Literal	Total	Annotators
Original(English)	MOH-X	315	332	647	-
Translation(Chinese)	Google	315	332	647*4	9
	Opus	315	332		
	Youdao	315	332		
	GPT-4o	315	332		
Translation(Italian)	Google	315	332	647*4	9
	Opus	315	332		
	Youdao	315	332		
	GPT-4o	315	332		

Table 4.4: *Overview of the multilingual metaphor translation evaluation (MMTE) dataset. The dataset includes 647 English source sentences (from MOH-X) evenly split between metaphorical and literal usages. Each sentence is machine-translated into Chinese and Italian using four different systems: Google Translate, Opus-MT, Youdao, and GPT-4o. Each translation is annotated by three out of nine native-speaking annotators per language.*

4.2.5 Annotation and Inter-annotator Agreement

To provide a clear overview of the constructed resource, we summarize the resulting corpus in Table 4.4 before proceeding to detailed analyses. The table reports the number of metaphorical and literal instances, their distribution across languages and translation systems, and the number of human annotators per language. This snapshot offers a concise view of the corpus scale and composition.

As described in Section 4.2.2, each language was annotated by nine annotators, divided into three groups, with each group independently annotating the entire dataset once. Within each group, instances were evenly distributed such that each annotator worked on a distinct subset of data. Consequently, every instance has three independent annotation scores and three post-edited translations. The final reference translation was then reviewed and selected by professional translators to ensure quality and consistency.

To assess annotation consistency, we report inter-annotator agreement using Krippendorff's α in Table 4.2.5. Overall, literal translations yield higher agreement scores across standard translation quality dimensions—Fluency, Intelligibility, Fidelity, and Overall—and Authenticity compared to metaphorical ones. This trend reflects

Krippendorff's α	Fluency	Intelligibility	Fidelity	Overall	Authenticity	Equivalence	Emotion
IT-Metaphorical	0.13	0.29	0.38	0.32	0.28	0.24	0.37
ZH-Metaphorical	0.15	0.25	0.32	0.36	0.31	0.19	0.34
IT-Literal	0.31	0.43	0.45	0.47	0.36	0.41	0.23
ZH-Literal	0.29	0.44	0.46	0.41	0.48	0.29	0.35

Table 4.5: *Inter-annotator agreement scores (measured by Krippendorff's α) across different evaluation criteria for metaphorical and literal translations in the MMTE dataset. Scores are reported separately for Italian (IT) and Chinese (ZH) translations, covering Fluency, Intelligibility, Fidelity, Overall Quality, Authenticity, Equivalence, and Emotion. Higher values indicate stronger agreement among annotators.*

.

the inherent ambiguity and interpretive variability associated with metaphorical language. These results underscore both the challenge and the value of fine-grained evaluation for metaphor translation.

4.3 Corpus Analysis

We conduct a comprehensive multi-perspective analysis of our multi-lingual corpus on metaphorical and literal expressions, considering multiple dimensions and different methods in §4.3. Firstly, we analyze the differences in translation quality between metaphorical and literal expressions in §4.3.1, highlighting the higher error rates and non-equivalent translations associated with metaphorical language. We then examine the correlations between the suggested fine-grained human evaluation protocol in §4.3.2, the correlations between Emotional and Metaphorical Expressions in §4.3.4, and the crucial role of Metaphor Equivalence in metaphor translation Quality Estimation (QE) in §4.3.5. Additionally, we analyse the translation quality between typologically different languages in §4.3.6. We also provide a case study indicating that translating between more typologically distant language pairs is harder, by comparing EN-ZH and EN-IT pairs in §4.3.7.

EN-ZH		Manual Evaluation Metrics				
		Fluency	Intelligibility	Fidelity	Authenticity	Overall
Google	Metaphorical	4.47	4.31	4.25	4.12	4.34
Google	Metaphorical (full)	4.75	4.73	4.72	4.64	4.71
Google	Literal	4.53	4.55	4.53	4.49	4.54
Opus	Metaphorical	3.87	3.52	3.39	3.22	3.59
Opus	Metaphorical (full)	4.40	4.32	4.32	4.25	4.32
Opus	Literal	3.93	3.80	3.74	3.75	3.82
Youdao	Metaphorical	4.67	4.59	4.53	4.53	4.60
Youdao	Metaphorical (full)	4.82	4.81	4.80	4.85	4.82
Youdao	Literal	4.66	4.67	4.65	4.62	4.66
GPT-4o	Metaphorical	4.05	4.25	4.35	4.05	4.17
GPT-4o	Metaphorical (full)	4.59	4.32	4.62	4.59	4.53
GPT-4o	Literal	4.54	4.54	4.17	4.22	4.37
EN-IT						
Google	Metaphorical	4.57	4.46	4.30	4.32	4.44
Google	Metaphorical (full)	4.78	4.77	4.72	4.63	4.73
Google	Literal	4.77	4.68	4.58	4.67	4.68
Opus	Metaphorical	4.45	4.29	4.14	4.16	4.29
Opus	Metaphorical (full)	4.78	4.77	4.74	4.63	4.73
Opus	Literal	4.65	4.53	4.45	4.52	4.54
Youdao	Metaphorical	4.36	4.16	3.96	4.04	4.16
Youdao	Metaphorical (full)	4.73	4.73	4.67	4.56	4.67
Youdao	Literal	4.53	4.42	4.29	4.38	4.41
GPT-4o	Metaphorical	4.41	4.34	4.14	4.25	4.28
GPT-4o	Metaphorical (full)	4.50	4.60	4.64	4.55	4.57
GPT-4o	Literal	4.59	4.55	4.50	4.55	4.55

Table 4.6: *Manual Metaphorical and literal expression evaluation averages, which employ a 5-point scale to assess the quality and characteristics of expressions. Metaphorical (full) refers to translations annotated as having full-equivalence.*

EN-ZH		**Automatic Evaluation Metrics**				
		BLEU1	BLEU4	Rouge-L	BERTScore	GPT-4o
Google	Metaphorical	0.58	0.20	0.62	0.765	4.44
Google	Metaphorical (full)	0.52	0.20	0.78	0.766	4.75
Google	Literal	0.73	0.38	0.76	0.768	4.67
Opus	Metaphorical	0.49	0.10	0.53	0.737	3.56
Opus	Metaphorical (full)	0.44	0.13	0.65	0.735	4.14
Opus	Literal	0.49	0.14	0.54	0.732	3.77
Youdao	Metaphorical	0.64	0.26	0.67	0.759	4.64
Youdao	Metaphorical (full)	0.53	0.23	0.82	0.764	4.74
Youdao	Literal	0.80	0.57	0.83	0.766	4.74
GPT-4o	Metaphorical	0.58	0.26	0.60	0.764	4.69
GPT-4o	Metaphorical (full)	0.64	0.30	0.68	0.765	4.87
GPT-4o	Literal	0.59	0.29	0.64	0.761	4.90
EN-IT						
Google	Metaphorical	0.50	0.22	0.60	0.811	4.51
Google	Metaphorical (full)	0.65	0.42	0.74	0.811	4.55
Google	Literal	0.68	0.47	0.74	0.807	4.68
Opus	Metaphorical	0.48	0.19	0.58	0.808	4.06
Opus	Metaphorical (full)	0.64	0.42	0.73	0.809	4.45
Opus	Literal	0.65	0.43	0.71	0.803	4.29
Youdao	Metaphorical	0.45	0.17	0.54	0.805	3.95
Youdao	Metaphorical (full)	0.61	0.38	0.69	0.801	4.34
Youdao	Literal	0.58	0.30	0.64	0.799	4.13
GPT-4o	Metaphorical	0.52	0.24	0.60	0.812	4.53
GPT-4o	Metaphorical (full)	0.59	0.27	0.67	0.810	4.85
GPT-4o	Literal	0.55	0.26	0.65	0.811	4.81

Table 4.7: *Automatic Metaphorical and literal expression evaluation averages, which **GPT** employ a 5-point scale to assess the quality and characteristics of expressions, whilst other **Automatic Evaluation Metrics** provide scores ranging 0-1. Metaphorical (full) refers to translations annotated as having full-equivalence.*

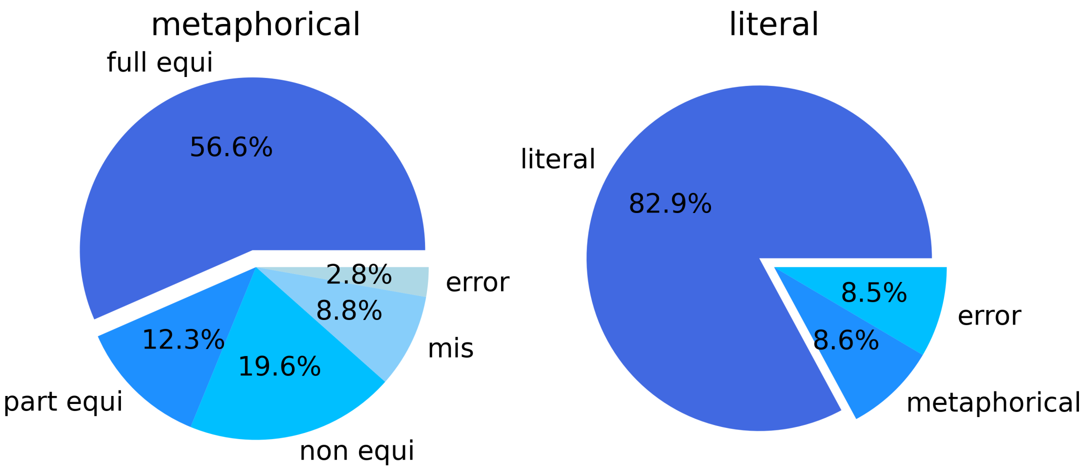


Figure 4.5: *Equivalence distributions of metaphorical and literal expression translations from annotators. non equi, part equi, and full equi refer to non-, part-, and full- equivalence, respectively. mis denotes mistranslation.*

4.3.1 Comparing the translation of Metaphorical vs Literal Expressions

As shown in Figure 4.5, the analysis of equivalence labels in metaphorical and literal expression translations highlights the varying degrees of equivalence and accuracy in translating metaphorical and literal expressions. The proportions shown in the pie charts represent the average across all human annotators and all languages in the corpus, providing a robust overall view of translation quality. Approximately 20% of metaphorical expressions are found to be translated without proper correspondence to the intended metaphorical meaning *(non-equi)*. Furthermore, more than 10% of metaphorical translations exhibit a failure to comprehend the intended metaphor or contain mistakes or inaccuracies (*mis* and *error*). These results emphasise the challenges associated with translating metaphorical expressions.

Table 4.6 & 4.7 presents scores from manual and automatic evaluations to compare the translation of metaphorical and literal expressions. It can be seen that translating metaphorical expressions poses greater difficulty compared to translating literal expressions. In both EN-ZH and EN-IT translation, the metaphorical expression translations generally obtained lower scores in all Manual Evaluation Metrics and GPT-4o compared to translations of literal expressions from the same MT system. Our automatic evaluation metrics also support this observation, with lower scores from BLEU1, BLEU4, and Rouge-L for metaphorical expression translations, suggesting a reduced level of similarity and alignment with reference translations.

Most importantly, we separately calculate the evaluation scores for full-equivalence translations. The results show that when metaphors are translated faithfully, their scores are significantly higher. This demonstrates that although translating metaphors is a challenging task, achieving the correct form of translation often results in more satisfactory outcomes, therefore highlighting the importance of having comprehensive translation evaluation metrics.

BERTScore struggles to distinguish the performance between metaphorical and literal translations. This limitation may be due to the methods relying on contextual embeddings and cosine similarity struggles to capture the subtle semantic differences inherent in metaphorical language. This highlights the need for specialised evaluation tailored to the complexities of metaphor.

4.3.2 LLMs Equivalence Assessment

	GPT-3.5[2]	**GPT-4o**[3]	**Gemini Pro**[4]
EN-IT full	86.0	86.7	85.7
EN-IT others	92.5	94.0	91.7
EN-ZH full	76.2	76.5	74.4
EN-ZH others	84.1	86.3	84.7

Table 4.8: *Accuracy of LLMs in classifying metaphor equivalence when compared to human annotations. full refers to translations annotated as having full equivalence, whilst others refers to translations as having non- or part- equivalence.*

As shown in Table 4.8, we employ LLMs to annotate the equivalence of metaphor translations and compare the results with the human-annotated reference data. The dataset used for the evaluation here is a final revised version as reference data. Following the evaluation workflow in Figure 4.1, previously error or misunderstanding translations were post-edited by annotators and reviewed by experts. The

[2]gpt-3.5-turbo-0125 https://platform.openai.com/
[3]gpt-4o-2024-05-13 https://platform.openai.com/
[4]gemini-1.0-pro-001 https://cloud.google.com/

corrected outputs were then integrated into valid categories (e.g., full-, part-, or non-equivalence) and used as updated reference data. The LLM-based evaluation results demonstrate a high level of consistency with human annotators. Moreover, we task LLMs with providing explanations for their annotations, offering insights into their interpretation of metaphorical content across different languages. For instance, consider the sentence pair: EN: "She swallowed the last words of her speech" and ZH: "她咽下了最后几句话." Here, "咽下" is a translation with full-equivalence. The explanation from GPT-4 is as follows: "*Both in the source sentence and the translation, 'swallowed' and '咽下' are used metaphorically to mean that she did not say the last words of her speech. The literal meanings of 'swallow' and '咽下' are also the same, referring to the action of making food or drink go from your mouth down through your throat and into your stomach.*" Detailed examples of these explanations can be found in §4.3.8. This comparison reveals that LLMs can effectively complement human efforts, providing reliable and insightful evaluations that are crucial for high-quality translation assessments at scale.

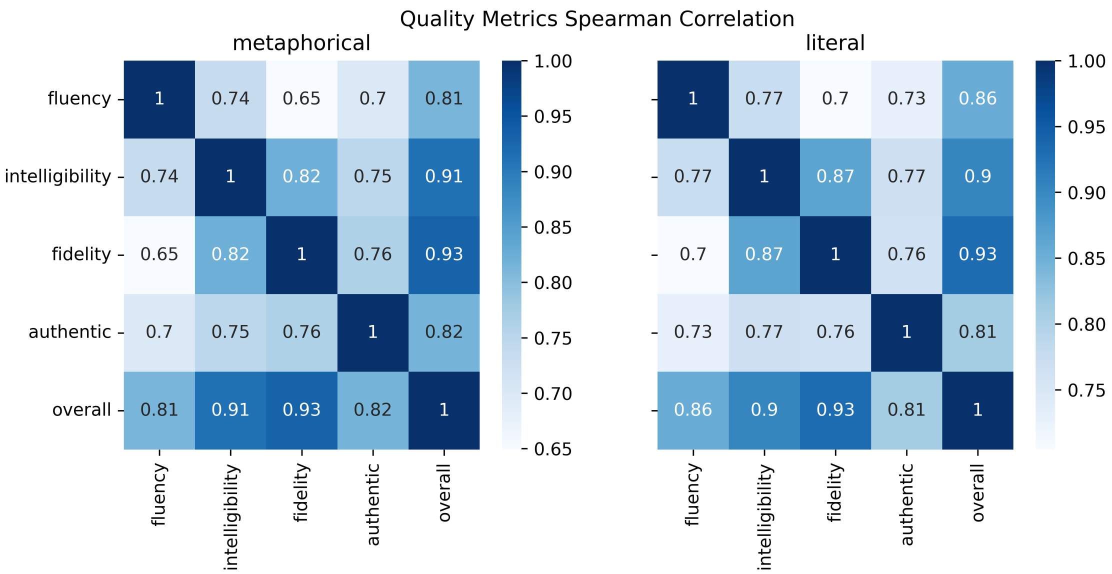


Figure 4.6: *Spearman's rho correlation heatmap of manual evaluation quality.*

4.3.3 Correlation Analysis of Fine-grained Human Evaluation Metrics

Figure 4.6 shows the fine-grained correlations between human evaluation metrics by calculating pairwise Spearman's rho correlations between the criteria of Fluency, Intelligibility, Fidelity, Authenticity, and Overall Score. We observe that Fluency, Intelligibility, Fidelity, and Authenticity are all strongly correlated with one another, suggesting that these dimensions are not entirely independent but reflect related aspects of translation quality. Moreover, all four criteria show very strong positive correlations with the Overall Score, confirming that these dimensions collectively contribute to human judgments of overall quality. Since Spearman's rho measures monotonic relationships without assuming normality, these findings align with the intuition that human annotators tend to rank translations consistently across different dimensions. Interestingly, the correlations between these metrics are slightly weaker in metaphorical cases compared to literal ones, indicating that in metaphor translation, evaluators treat each quality dimension more independently, likely reflecting the additional complexity and subjectivity of translating metaphors.

4.3.4 Correlation Analysis of Emotion and Equivalence Metaphor

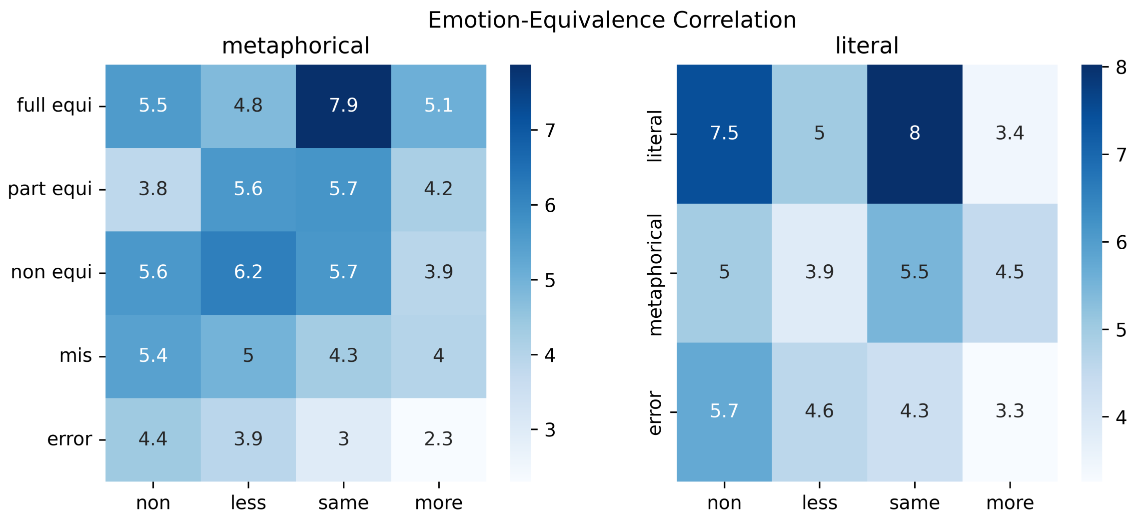


Figure 4.7: *Emotion-Equivalence correlation heatmap based on co-occurrences.*

We investigate the correlation between the degree to which emotional content is preserved in translation and the translation of figurative expressions in Figure 4.7. The figure presents a correlation heatmap based on the logarithm of co-occurrence counts between Emotion labels and Equivalence categories, where a co-occurrence is defined as an annotated translation instance that is annotated both an Emotion label e and an Equivalence category q. Formally, each cell in the heatmap is computed as:

$$C(e, q) = \log\big(N(e, q) + 1\big), \tag{4.6}$$

where $N(e, q)$ denotes the number of translation instances in which Emotion label e and Equivalence category q co-occur.

It is noticeable that emotion levels perceived by annotators tend to remain constant if the original metaphorical expression is translated to a fully equivalent version, and the original literal expression is translated to a literal version. This observation indicates that maintaining the figurative status of translations is a reasonable strategy for keeping the emotional expression authentic. For example, the metaphorical expressions "*swallow the sentence*" and "咽下这句话" both convey reluctance, whilst the Chinese literal translation "没说这句话" does not. We also observe that non-equivalent translations tend to keep little of the emotion contained in the original metaphorical expressions. In contrast, fully equivalent translations tended to retain more of the original emotion, while partially equivalent translations were about equal in retaining the same and conveying less emotion. This finding highlights the difficulty of maintaining emotional content during translation and show how the importance of equivalence are in retaining the emotion of metaphorical expressions.

4.3.5 Impact of Metaphor Equivalence

Besides maintaining emotional salience, fully equivalent metaphor translations and literal translations of literal expressions demonstrate higher translation quality. This is revealed in Figure 4.8, which shows that fully equivalent translations of metaphorical expressions outperform others in the dimensions of Fluency, Intelligibility,

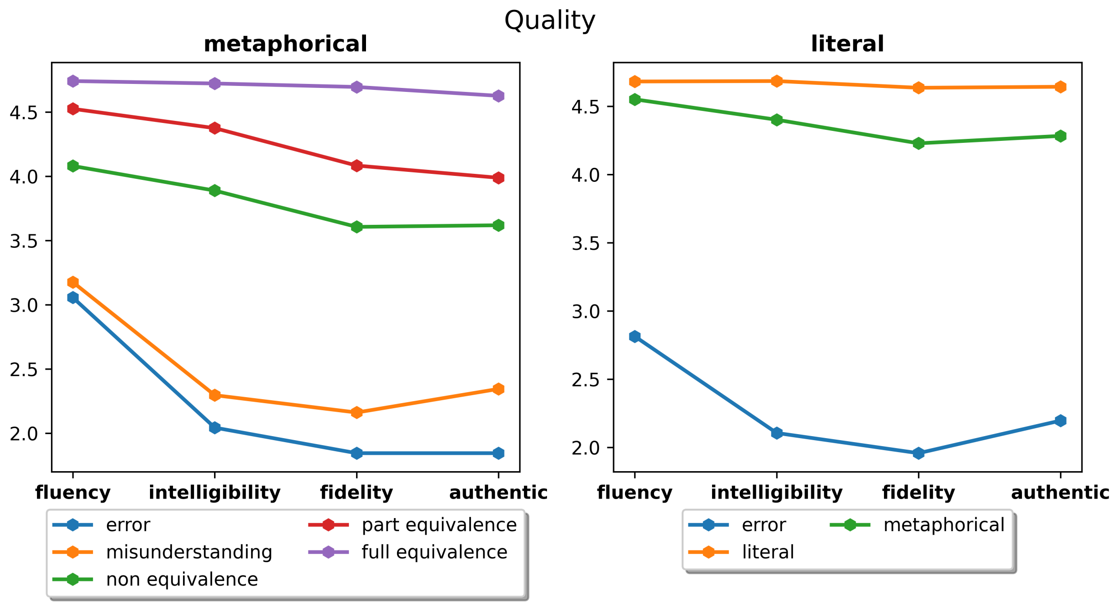


Figure 4.8: *Average quality scores of manual evaluation of metaphorical and literal expression translation.*

Fidelity, and Authenticity, whilst literal translations of literal expressions outperform other versions in the four dimensions. We also observe that part-equivalent and non-equivalent translations of metaphors cause more severe translation quality degradation than metaphorical translations of literal expressions. We hypothesise that literal translations of metaphorical expressions between languages spoken by different communities result in unnatural literal statements, which also supports the observation that the *translation of metaphorical expressions is harder than that of literal expressions.*

4.3.6 Impact of Different Language

Figure 4.9 presents a comparison of the average evaluation scores of EN-ZH and EN-IT translations across all models. The results show that the average translation quality is lower for Chinese compared to Italian, despite both being translated from English. This can be attributed to several factors. Firstly, Chinese and English belong to different language families and possess distinct linguistic structures, with the grammatical disparities posing challenges for accurate translation. Secondly, cultural differences also play a significant role in translation quality. Translating metaphors accurately requires a deep understanding of cultural nuances and idiomatic expressions between the source and target languages. Failure to grasp these nuances can lead to mistranslation or loss of the intended meaning. Furthermore, the availability and quality of language resources and machine translation models differs for Chinese and Italian.

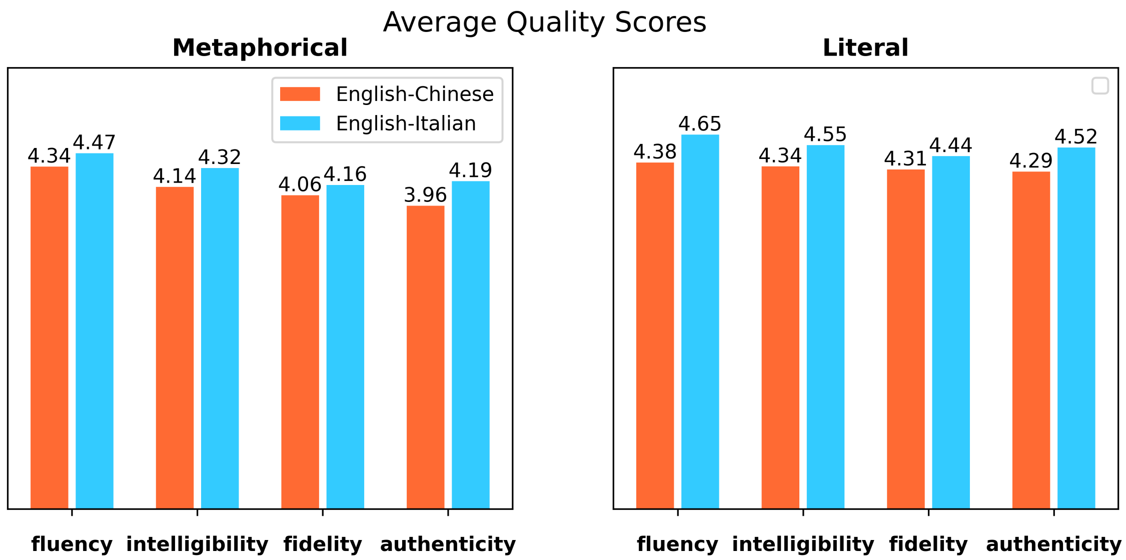


Figure 4.9: *Average quality scores of manual evaluation for EN-ZH and EN-IT.*

4.3.7 Influence of Linguistic Typology on Translation Difficulty

Linguistic typological features are known to be able to assist translation and rank candidates for multilingual transfer (Oncevay et al. 2020). The experimental results of Opus in Table 4.6 & 4.7 support a similar conclusion, that translation between a language pair with a closer typological relationship (EN-IT) is easier than a more distant pair (EN-ZH). However, this conclusion does not hold for the experimental results of Google and Youdao in Table 4.6 & 4.7. Youdao, a popular commercial multilingual translation tool in the Chinese community, achieves better translation performance in the EN-ZH direction than EN-IT. We hypothesize that the size of the corpus is much more important for translation quality compared to linguistic typology. Due to the above observations, the potentially larger EN-ZH parallel corpus that Youdao and Google have compared to EN-IT, and the relatively balanced sizes of EN-ZH and EN-IT parallel corpora that Opus holds, may aid in explaining the observed difference.

4.3.8 Metaphor Explanation with LLMs

Prompts	Please compare the source sentence and its translation, and determine to what extent the @target word@ and its translation are Equivalent in figuration. Here are the definitions of the three types of Equivalence and two types of mistake: Full-Equivalence: When comparing the source sentence and translation sentence, both the literal meanings and the contextual meanings of the target word are the same. Example: source: He @injected@ new life into the performance. translation: 他给表演@注入@了新的生命Part-Equivalence: When comparing between the source sentence and translation sentence, only the contextual meanings of the target word are similar. The literal meaning of the target word between the source sentence and translation are different, but they are both metaphorical. Example: source: @Wallow@ in your success! translation: @沉浸@在你的成功中吧！Non-Equivalence: When comparing the source and translation, only the contextual meanings of the target word are similar. However, the translation is a non-metaphorical expression, making the literal meaning of the target word between the source and translation different. Example: source: Sales were @climbing@ after prices were lowered. translation: 价格下跌后销售额@上升@。Misunderstanding: When the literal meanings are similar between the target word in the source text and the target word in the translation, but the translation conveys no contextual meaning like the target in the source language. Example: source: I @attacked@ the problem as soon as I got out of bed. translation: 我一下床就@攻击@了问题Error: When the target word is mistranslated, meaning not only that the contextual meanings are different between the source and translation, but their literal meanings also differ. Example: source: The @stamp@ fruit extract the juice. translation: 果果提取果汁。
Query	Source: **English source sentence** Translation: **Target language translation** Equivalence and Explanation:

Table 4.9: *Prompt and Query Format for LLMs in Evaluating Metaphor Translation Equivalence and Generating Explanations.*

We used LLMs to annotate metaphor equivalence and attempted to guide the models to provide explanations for their evaluations. LLMs, particularly GPT-4, demonstrate an understanding that approaches human annotators in terms of both semantic and rhetorical comprehension. The specific formats of the prompts and queries are shown in the Table 4.9. By providing specific examples and explanations for each type of equivalence and including them in the prompts, we aimed to give LLMs references for comparison.

As shown in Table 4.10, LLMs showcased a robust ability to understand and interpret metaphors, providing comprehensive explanations that covered both se-

	EN: The news were leaked to the paper. **ZH**: 消息泄露 给了报纸。	**EN**: The colors don't harmonize. **IT**: I colori non si armonizzano.
GPT-3.5	Both the source sentence and the translation use a metaphorical expression to convey the same meaning. The target words "leaked" and "泄露" both literally and contextually mean that information was disclosed or released unintentionally or secretly.	The source sentence and the translation both use the word "harmonize" (and its Italian equivalent "armonizzano") in the same context, with both the literal and contextual meanings being the same.
GPT-4	The literal and contextual meanings of "leaked" in the source sentence and "泄露" in the translation are the same. Both terms refer to the unauthorized release or disclosure of information.	The source sentence and the translation have both the same literal and contextual meanings of the target word "harmonize". In both languages, it refers to the act of being in harmony or agreement.
Gemini Pro	**Literal Meaning:** Both "leaked" and "泄露" share the literal meaning of confidential information being unintentionally or secretly revealed. **Contextual Meaning:** Both words fit the context of the sentence, implying an act of disclosing information without authorization. The information flow (news to the paper) is consistent. Therefore, both the literal and contextual meanings are preserved in the translation, making it a case of Full-Equivalence.	The word "harmonize" and its Italian translation "armonizzano" share both literal and contextual meanings. * **Literal Meaning:** Both words refer to the concept of elements coming together in a pleasing and balanced way.* **Contextual Meaning:** In the context of colours, both words imply that the colors used together do not create a visually appealing or cohesive effect. Therefore, the translation accurately conveys the meaning of the source word, making it a case of full equivalence.

Table 4.10: *Examples of explanations given by the LLMs for metaphor translations, selecting one full-equivalence translation each in Chinese and Italian.*

mantic nuances and rhetorical aspects. This performance indicated a high level of competency in handling cross-linguistic tasks.

By analyzing the explanations provided by LLMs, we were able to validate their effectiveness in metaphor translation tasks. This analysis demonstrated that LLMs could not only understand and interpret metaphors accurately but also articulate the reasoning behind their evaluations. This capability is crucial for ensuring that the subtleties of metaphorical language are preserved in translation. LLMs offered reliable and insightful evaluations that are essential for high-quality translation assessment. Their ability to generate detailed and contextually accurate explanations for their decisions highlights their potential as a robust tool in the translation process.

4.4 Conclusion

MMTE is the first work to systematically investigate how translations are affected by metaphor in a fine-grained and multi-lingual setting. MMTE also introduces Equivalence as a new dimension of metaphor translation evaluation and verifies its relationship with emotional salience and translation quality. Moreover, we conducted thorough experiments on the proposed evaluation dimensions and verified the increased difficulty of translating metaphorical expressions compared to literal expressions.

We additionally release MMTE, a high-quality metaphor translation corpus, through ORDA[5], the University of Sheffield's research data repository. This resource is designed to support future work on developing automatic evaluation metrics for metaphor translation. Future work intends to combine MMTE with additional well-designed automatic metrics aligning with specific human evaluation dimensions proposed in the paper.

[5]The dataset can be downloaded via this link: https://figshare.com/s/bd4137fb3a05cf122b01.

Chapter 5

Exploring Task Performance with Interpretable Models via Sparse Auto-Encoders

The interpretability challenge in Large Language Models (LLMs) becomes particularly acute in metaphor detection, where polysemantic neurons—activated by multiple unrelated linguistic features—fundamentally hinder model transparency. Traditional metaphor detection frameworks like MelBERT and RoPPT rely on external linguistic structures (e.g., parse trees or frame semantics) to filter noise, yet fail to address the core issue of entangled internal representations. This chapter introduces a novel paradigm that bridges mechanistic interpretability with task performance by decomposing LLM activations into sparse, monosemantic features.

Central to our approach is the application of sparse auto-encoders to disentangle polysemantic neurons in LLMs. By training a dictionary learning algorithm on model activations, we extract interpretable feature sets that correspond to distinct semantic categories (e.g., literal vs. figurative meanings). This decomposition enables precise identification of ambiguous metaphorical targets through feature activation analysis. For instance, when processing "The champagne flowed at the wedding," our method isolates conflicting activations: dominant features linked to literal fluid movement versus secondary activations tied to celebratory abundance.

We further develop an adaptive enhancement mechanism that selectively reformulates ambiguous metaphors using GPT-4-generated clarifications. Unlike conventional input rewriting, our system triggers interventions only when feature misalignments exceed a learned threshold, preserving model autonomy while resolving critical ambiguities. Evaluations on MOH-X and TroFi datasets demonstrate significant improvements: Llama-3 achieves an absolute gain of 4.3% in metaphor detection accuracy, outperforming RoPPT by 4.2% while requiring no task-specific architectural modifications.

Key innovations include: **Feature-activation diagnostics**: Quantifying metaphor ambiguity through the divergence between literal and figurative feature clusters in activation space. **Minimal-intervention reformulation**: Preserving original input structure while injecting targeted disambiguating cues (e.g., "flowed implies plentiful availability") based on activation patterns. **Cross-model generalizability**: Consistent accuracy gains (3.8–5.4%) across diverse LLMs (Mistral, Phi-3) without model retraining.

This chapter establishes that interpretability methods need not trade performance for transparency. By directly addressing the root cause of metaphor misunderstanding—polysemantic neuron activations—our approach simultaneously enhances both model interpretability and task capability, advancing toward LLMs that explain their reasoning while refining it.

5.1 Related Work

Interpretability of LLMs Belinkov & Glass (2019) conducted a review of interpretability methods in neural language models, emphasizing the importance of disentangling features for a clearer understanding of LLMs' internal structures. Interpretability techniques such as attention visualization (Karpathy et al. 2015, Qian et al. 2016, Liu et al. 2018, Abnar & Zuidema 2020), layer-wise relevance propagation (Gupta & Schütze 2018, Maladry et al. 2023, Song et al. 2024), and probing classifiers (Conneau et al. 2018, Belinkov 2022, Pantazopoulos et al. 2024) offer partial insights but fall short in fully explaining the model's internal decision-making processes. Sparse representations (Faruqui et al. 2015) and feature decomposition (Murdoch et al. 2018), in contrast, show more promise in producing human-readable interpretations of model activations.

Mechanistic interpretability aims to decompose neural networks into their fundamental components to understand their behavior at a granular level. One of the primary challenges in this approach is polysemanticity—the phenomenon where individual neurons respond to multiple, unrelated inputs (Olah et al. 2020). This complicates attempts to assign clear functions to specific neurons, as a single neuron might represent different linguistic constructs depending on the context. Olah et al. introduced the concept of circuits in LLMs, demonstrating how understanding neuron activations and their interactions can reveal more about how models process certain inputs. They proposed that by breaking down LLMs into circuits of interacting neurons, we could begin to form a clearer picture of model behavior.

Research on toy models of Elhage et al. (2022) suggests that polysemantic neurons arise from the network's need to efficiently compress features into fewer neurons than there are distinct features in the data. They demonstrated that superposition, a mechanism where neurons represent multiple independent features, complicates interpretability. To address this, the authors conduct experiments on single-layer toy models using dictionary learning and sparse feature extraction to decompose the neuron activations into more interpretable units.

As disscussed in §2.2.2, sparse representations are a promising technique for tackling polysemanticity and superposition. In sparse autoencoders, a type of dictionary learning algorithm, features are represented with minimal overlap between neurons, making it easier to interpret their activations. This method enables a more monosemantic representation of features, meaning each neuron responds to fewer, more clearly defined patterns. Cunningham et al. (2023) explored this idea to generate interpretable features from LLM activations. Their approach allows for more precise identification of the features that cause counterfactual behavior in tasks like indirect object identification.

Another aspect of interpretability involves applying these techniques to specific downstream tasks, such as mathematical reasoning or metaphor detection, where understanding model behavior is critical to improving performance. Doshi-Velez & Kim (2017) called for more rigorous interpretability studies in these specific contexts, arguing that the demand for transparency is particularly high when models are used in complex and sensitive domains. In response, our work applies sparse decomposition methods to understand how LLMs handle logical reasoning in mathematics or process ambiguous language in metaphor detection. These applications show the practical benefits of interpretability in improving both model performance and trustworthiness.

As discussed in §2.2.2, metaphor interpretation can be approached through property extraction, word-level paraphrasing, and explanation pairing. Our SAE-based approach can be seen as complementary to these categories, as it focuses on uncovering latent semantic features in model activations that influence how target words are interpreted. By combining sparse autoencoder feature extraction with LLM-based natural language explanation (inspired by OpenAI's auto-interpretability techniques), we build a bridge between mechanistic interpretability research and metaphor interpretation. This connection motivates the methodology developed in the following sections, where we use these tools not only to expose what representations a model uses for metaphorical targets but also to guide potential interventions when the model misinterprets them.

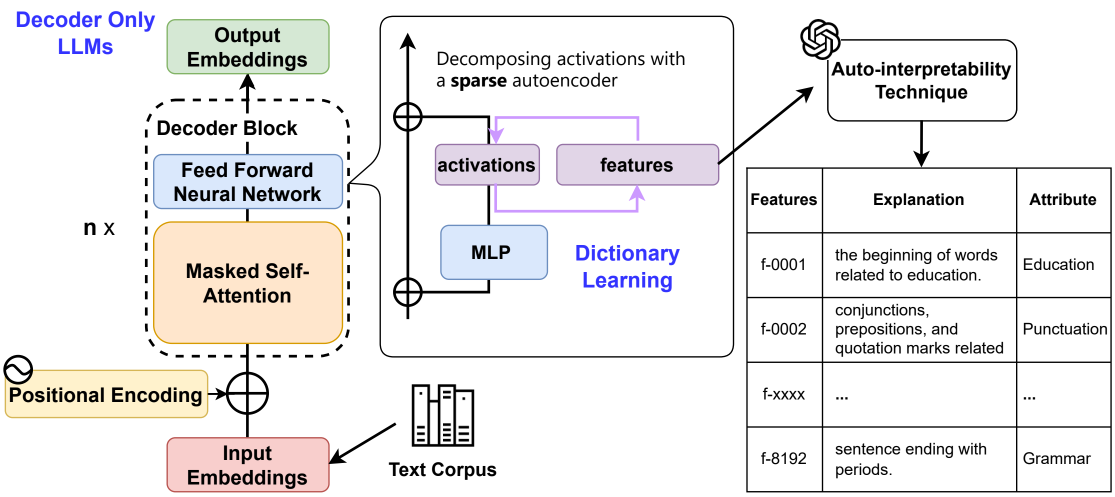


Figure 5.1: *Framework of decomposing LLMs by Dictionary Learning. The framework extracts features from LLMs using a sparse autoencoder to isolate monosemantic features from polysemantic neurons. Once trained, OpenAI's auto-interpretability techniques (Bills et al. 2023) prompt GPT-4 to verbally describe the meaning of these features.*

5.2 Methodology

Our experimental work is divided into two parts. In the first part, we employ dictionary learning to decompose LLMs and extract relatively monosemantic features, annotating these features using an auto-interpretability technique to produce human-readable dictionaries. In the second part, we use the explicit dictionary obtained from the first step along with a sparse autoencoder to analyze downstream tasks. We designed and implemented an automated pipeline to complete this step.

5.2.1 LLM Decomposition

In this step, we apply a sparse auto-encoder, a dictionary learning algorithm, to generate features from several open-source LLMs that provide a more monosemantic unit of analysis than individual neurons. Our approach is grounded in extensive prior research, particularly in the use of dictionary learning and related techniques for analyzing neural network activations, as well as broader literature on disentanglement (Olshausen & Field 1997, Lee et al. 2006, Yun et al. 2021, Elhage et al. 2022, Bricken et al. 2023, Cunningham et al. 2023).

Sparse Autoencoder To address neuron superposition in LLMs, we leverage sparse autoencoders (SAEs) to decompose activations of the feed forward neural network in the final decoder block into a set of disentangled, interpretable features. Concretely, we assume that an observed activation vector $\boldsymbol{x}$ can be represented as a sparse linear combination of an unknown set of latent vectors G, i.e.,

$$\mathbf{x} = \sum_{i=0}^{n_g} \mathbf{a}_i \mathbf{g}_i \tag{5.1}$$

where $\boldsymbol{g_i} \in G$ are the basis vectors and $\boldsymbol{a} = (a_0, ..., a_{n_g})$ is a sparse coefficient vector. n_g, the number of vectors in the set G, is significantly larger, even several times greater, than the dimension of the vectors $\boldsymbol{x}$. Each coefficient $\boldsymbol{a_i}$ indicates the contribution or activation strength of the corresponding basis vector $\boldsymbol{g_i}$ in reconstructing $\boldsymbol{x}$ with most $\boldsymbol{a_i} = 0$ due to the sparsity constraint.

In our case, $\boldsymbol{x}$ represents the activation values from one of the MLP layers of an LLM, whilst the dimension of $\boldsymbol{x}$ is the number of neurons in this MLP layer, which we denote as n_a. The unknown vector set G represents the ground truth we aim to learn. Our goal is to learn a dictionary of vectors F, referred to as dictionary features, using F to approximate the unknown vector set G, such that ultimately $\boldsymbol{f_i} \approx \boldsymbol{g_i}$.

To learn the dictionary F, we train a sparse auto-encoder with one single hidden layer. The dimension of the input and output is the same as n_a (i.e., dimension of $\boldsymbol{x}$) and the dimension of the hidden layer is the same as n_g (i.e., the number of dictionary features). Therefore, the entire auto-encoder can be divided into two components: the encoder (as shown in Equation 5.2), and the decoder (as shown in Equation 5.3). To ensure symmetry between encoding and decoding, tied weights are used in our auto-encoder. By making the decoder's weight matrix the transpose of the encoder's, the decoder can directly reverse the encoding process. This symmetry helps the model more accurately reconstruct the input by maintaining consistency between the two stages.

$$\mathbf{s} = \mathrm{ReLU}(F(\mathbf{x} - \mathbf{b}_d) + \mathbf{b}_e) \tag{5.2}$$

$$\hat{\mathbf{x}} = F^T\mathbf{s} + \mathbf{b}_d \tag{5.3}$$

where $\boldsymbol{b_e}$ and $\boldsymbol{b_d}$ are the biases of the encoder and decoder, respectively. The input is first mean-centered by subtracting the decoder bias $\boldsymbol{b_d}$, which represents the average activation of the input dimension learned during decoding. This ensures that the encoder learns sparse codes relative to the mean input, preventing the bias term from being redundantly encoded into the sparse representation.

The vector $\boldsymbol{s}$ is the sparse code produced by the encoder and serves as the model's estimate of the sparse coefficient vector $\boldsymbol{a}$. Each component s_i corresponds to the activation of a specific dictionary feature (i.e., a column of F). Due to the sparsity enforced by the architecture and the ReLU activation, only a small subset of these features become active for any given input. Thus, $\boldsymbol{s}$ provides an interpretable, disentangled representation of the underlying latent factors thought to compose the original activation pattern.

The vector $\hat{\boldsymbol{x}}$ is the reconstructed activation, obtained by linearly combining the dictionary features according to the sparse code $\boldsymbol{s}$ and then adding back the mean. A well-trained autoencoder satisfies $\hat{\boldsymbol{x}} \approx \boldsymbol{x}$, indicating that the learned sparse features successfully capture the essential structure in the original LLM activations.

The main goal of training is to minimise the following loss function:

$$\mathcal{L} = \underbrace{\frac{1}{|X|} \sum_{\mathbf{x} \in X} \|\mathbf{x} - \hat{\mathbf{x}}\|_2^2}_{\text{Reconstruction loss}} + \underbrace{\lambda \|\mathbf{s}\|_1}_{\text{Sparsity loss}} \tag{5.4}$$

The total loss function $\mathcal{L}$ of the autoencoder consists of two key components: the reconstruction loss and the sparsity loss. The ***reconstruction loss***, presented by Equation 5.4, measures the mean squared error between the input $\boldsymbol{x}$ and its reconstruction $\hat{\boldsymbol{x}}$, ensuring that the autoencoder accurately reconstructs the original input. The second component, $\lambda \|s\|_1$, is the ***sparsity loss***, which applies a regularization to the hidden activations $\boldsymbol{s}$, encouraging sparsity in the learned representation. The hyperparameter λ controls the trade-off between minimizing reconstruction error and enforcing sparsity in the latent space, helping the autoencoder discover a more compact and interpretable representation of the input data. Together, these two losses balance accurate reconstruction with a sparse, meaningful representation.

Auto-interpretability Technique Complementing SAEs, OpenAI's **auto interpretability** (Bills et al. 2023) techniques leverage large language models such as GPT-4 to automatically generate natural language explanations of neuron behavior. Instead of relying on time-consuming manual inspection, this approach provides each neuron of the model which needs to be explained with its most highly activating input examples, prompting GPT-4 to infer a human-readable description of the neuron's function.

These automatically generated explanations enable researchers to understand model internals at scale, offering insight into what features the network has learned such as: detecting specific domains, topics, or syntactic patterns, without requiring human annotation for every neuron. Importantly, this methodology provides a systematic way to generate human-readable feature descriptions, which we leverage to better interpret the semantic behavior of target words in context.

As shown in Figure 5.1, after training the sparse autoencoder, we adapt this technique to automatically interpret each learned dictionary feature in two steps. We first sample the activating text spans with the top activation from the training set and prompt GPT-4 to generate concise, human-readable descriptions of the semantic or syntactic patterns encoded by each feature. We provide GPT-4 with a few illustrative examples by using predefined semantic categories (e.g., Education, Punctuation, Grammar) in the prompt, and also provide the full set of these predefined semantic categories. Then we ask it to generate the most appropriate category for each feature based on these examples. This approach ensures stylistic consistency while allowing flexibility, reducing the risk of misclassification due to missing or overly rigid predefined labels. These categorical attributes are stored alongside the natural-language description and form an indexable representation of the feature space. This enriched annotation step enables downstream retrieval and alignment: during metaphor interpretation, the activated features can be compared with context by matching both their textual descriptions and their attribute labels, providing a method to determine which features are semantically relevant.

5.2.2 Downstream Application

Metaphor Interpretation As mentioned above, our process begins with the LLM autoencoder (SAEs augmented LLM) that is trained to decompose dense neuron activations into a set of sparse, human-interpretable features. Training is conducted on a large mixed corpus to ensure that our model retains strong versatility; further details about the corpus are provided in §5.3.3. After training, the SAE acts as a plug-in module that transforms a dense activation vector into a sparse feature vector, where each dimension corresponds to a learned dictionary feature.

The whole pipeline is shown in Figure 5.2. The first step applies this transformation to extract the activated features of the target word in the input sentence. To assess whether the model's activation aligns with the intended contextual sense, we

Metaphor Query	Target Word	Explanation
The champagne flowed at the wedding. Is the target word 'flowed' a metaphorical or literal expression?	flowed	1: terms related to movement or state change of liquid. **2: phrases related to social gathering or celebrations**. 3: rhythmic movement of a musical composition.
Math Query	**Symbol**	**Explanation**
If $\|4x+2\| = 10$ and $x < 0$, what is the value of x?	$\|\|$ (absolute value)	1: code language, including both programming and math functions. **2: numerical and mathematical expressions or symbols.** 3. URLs, hashtags, and alphanumeric characters.
	$<$ (less than)	1: punctuation, especially commas and hyphenated numbers, and discourse markers in potentially complex syntax structures such as order, sequence and list. 2: : words and phrases related to personal experiences or events. **3 special symbols and numerical values.**

Table 5.1: *Explanation of activated features for mathematical symbols and metaphorical terms in queries. The activated features, which are associated with mathematical attributes or the correct meanings of the metaphorical language, are in **bold**.*

rank activated features by the semantic similarity between them and the sentence context using GPT-3.5-turbo. Specifically, after obtaining the activated features of the target word, we follow the auto-interpretation procedure illustrated in Figure 5.1 to retrieve natural language explanations for each feature and the predefined semantic attributes associated with them. These feature explanations, together with the original sentence context, are then provided as input to GPT-3.5-turbo using a structured prompt (detailed in §5.2.3). The model is instructed to analyze the overall semantic meaning of the context and to rank the activated features according to their semantic similarity and relevance to the context. The returned ranked list highlights which features are most aligned with the contextual meaning, allowing us to detect whether the top-activated feature corresponds to the most contextually aligned feature. If not, we treat the target as potentially ambiguous. Using GPT-3.5-turbo as an independent evaluator of GPT-4–generated content is intended to avoid a self-evaluation bias, where the same model might implicitly validate its own generated explanations. By doing this, we reduce the risk of sys-

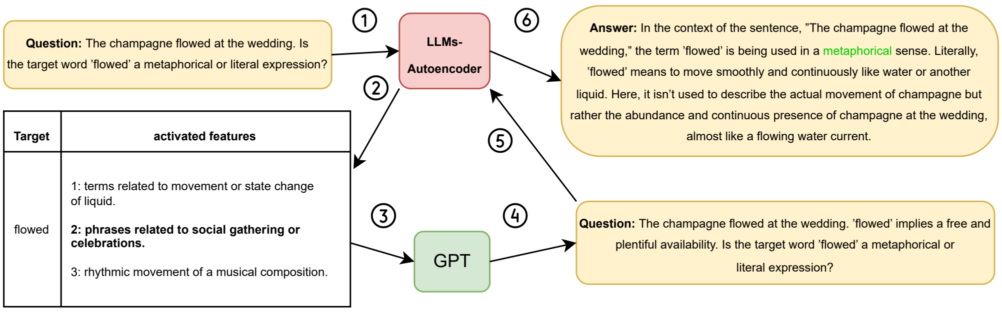


Figure 5.2: *Overview of the pipeline for handling ambiguity in metaphor detection/interpretation.(1) Activated features of the target word are extracted using an LLM with a trained sparse autoencoder plug-in. (2) The decomposed semantic features are compared via similarity matching to identify the most semantically relevant features for the target word. (3) Ambiguity detection is performed: if the most relevant feature is not ranked first, the target word is flagged as potentially misunderstood. GPT-4 is then prompted to generate a concise semantic clarification or contextual explanation for the target word. (4) The generated explanation is concatenated with the original sentence to form an augmented input. (5) The augmented input is re-fed into the original model. (6) This process yields a more accurate metaphor interpretation by correcting the potential understanding biases of the model, and enhances the metaphor detection task.*

tematically overestimating semantic alignment and ensure a more conservative and robust assessment.

In these cases, we invoke GPT-4 with a structured prompt (see §5.2.3) that includes the sentence, the target word, and the top-k activated features. GPT-4 generates a concise semantic clarification for the target word, which is then concatenated after the original sentence to form an augmented input. For example, when the word flowed is used metaphorically to express abundance, the augmented input begins with the original sentence, followed by the clarification "*'flowed' implies a free and plentiful availability.*". This strategy provides explicit semantic guidance while minimizing unnecessary modifications to the input.

The augmented sentence is re-fed into the model, which now processes it with its semantic clarification. The system then outputs both an interpretation of the target word's sense and a prediction of its metaphoricity. By correcting cases where the model's internal activations are dominated by irrelevant features, this pipeline improves the accuracy of metaphor interpretation and strengthens token-level metaphor detection, as demonstrated by the experimental results in §5.4.1.

Regarding GPT-4's role in reformulation, our approach does not simply rely on it as an input rewriter but leverages it as a knowledge distillation source, providing explicit semantic guidance. Furthermore, GPT-4 is incorporated based on automated pipeline we mentioned above, meaning that its intervention occurs only when the LLMs-autoencoder predicts a high likelihood of misinterpretation. This design ensures that GPT-4's involvement is both necessary and minimal, rather than being universally applied. Thus, our method does not depend on GPT-4 as the sole source of supplementary information but rather utilizes it as an external knowledge support to enhance LLM performance in high-ambiguity contexts.

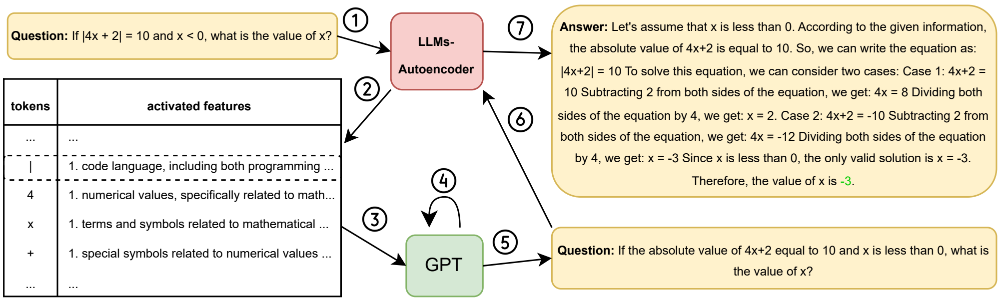


Figure 5.3: *Overview of the pipeline for handling ambiguity in mathematical problem-solving. (1) Activated features are extracted using LLMs with a trained autoencoder plug-in. (2) Rule-based methods analyze token activations to detect and label ambiguous mathematical symbols. (3) GPT-4 is employed to explain these symbols and rephrase the entire question. (4) The original and rephrased questions are compared for equivalence, and if necessary, the question is regenerated. (5) Correctly rephrased question is obtained. (6) The correctly rephrased question is re-input into the LLM. (7) This process results in an enhanced, more accurate answer.*

Mathematical Reasoning As seen in Table 5.1, in many cases, neither the primary nor secondary activations correspond to features from the mathematical domain. LLMs assign higher importance to unrelated linguistic or contextual associations when interpreting certain mathematical symbols (Srivatsa & Kochmar 2024). This misalignment introduces ambiguity in processing mathematical problems, leading to potential inaccuracies in understanding and solving such problems. As a result, this issue poses challenges for LLMs in effectively handling tasks involving mathematical reasoning.

Our approach involves a multi-step process to enhance the understanding and answering of ambiguous mathematical questions. First, we employ a rule-based method that systematically analyzes the activated features of each token by mapping them to predefined mathematical categories and identifying inconsistencies. This process begins by extracting the top activated features for each mathematical symbol and comparing them against a curated set of features of mathematical attributes. As shown in Table 5.1, several symbols (e.g., $||$, $<$) display primary activations that correspond to non-mathematical concepts (such as code language or punctuation). If a symbol's dominant features do not align with known mathematical attributes, it is flagged as ambiguous. Next, GPT-4 is employed to automatically explain these ambiguous symbols within the original problem and rephrase the entire question for clarity.

Once rephrased, the original and modified questions are compared to verify their equivalence by Gemini 1.0-Pro, a language model of comparable capability to GPT-4, ensuring robustness and reducing reliance on a single model. If the two versions are not equivalent, the rephrasing process is repeated to generate a valid rephrased question. Once a correctly rephrased question is obtained, it is then input into the model, resulting in a more accurate and enhanced answer. This process ensures better handling of ambiguity and improved model performance.

We selected mathematical reasoning and metaphor detection as downstream tasks because they test different but crucial capabilities of language models. Mathematical reasoning emphasizes precise logical thinking and structured problem-solving, while metaphor detection challenges the model's ability to understand abstract figurative language and context. These tasks are complementary, with one focusing on logical precision and the other on semantic adaptability, making them important and representative benchmarks for evaluating diverse aspects of language model performance.

5.2.3 GPT prompts

To ensure reproducibility and clarity of our approach, we explicitly document the LLMs prompt templates used in both the metaphor interpretation and mathematical reasoning pipelines. These prompts are designed to be modular, interpretable, and easily adaptable to future studies.

Prompt	You are an expert in figurative language interpretation and semantic analysis. Given a sentence, a target word and a list of activated semantic features for that word, each with its natural-language explanation and predefined semantic attributes. Your task is to analyze the sentence and determine the intended meaning of the target word in this context and rank the given features in descending order of their semantic similarity to the intended meaning of the target word. Please follow these instructions carefully: 1. Return your results as a ranked list from most relevant to least relevant. 2. Do not modify the original features and their explanation, just rank. Return your answer strictly in the following JSON format: { "ranked_features": [{ "feature": "feature id 1", "explanation": "explanation", "attribute": "attribute", "relevance_score": "0-1 score" }, { "feature": "feature id 2", "explanation": "explanation", "attribute": "attribute", "relevance_score": "0-1 score" }] }
Query	**Sentence**: sentence **Target Word**: target_word **Activated Features**:feature_list

Table 5.2: *Prompt and Query Format for GPT-3.5-turbo to rank the feature in Metaphor task pipeline.*

Prompt	You are an expert in figurative language understanding. Given a sentence, a target word within that sentence, and the top-k activated semantic features extracted from a sparse autoencoder trained on a large language model, your task is to generate a concise semantic clarification of the target word that resolves potential ambiguity. Please follow these instructions carefully: 1. Analyze the target word's meaning in the given sentence context. 2. Compare the meaning with the list of semantic features provided. 3. Generate a short clarification (1–2 sentences) that specifies the intended sense of the target word. 4. Do NOT rewrite the whole sentence. Your output should be a clarification only. Return your answer in the following JSON format: { "target_word": "target word", "clarification": "1–2 sentence explanation of intended meaning" }
Query	**Sentence**: {sentence} **Target Word**: {target word} **Top-k Activated Semantic Features**: {feature list}

Table 5.3: *Prompt and Query Format for GPT-4 to augment the input in Metaphor task pipeline.*

In the metaphor interpretation pipeline, Firstly, we use GPT-3.5-turbo to perform feature ranking and ambiguity detection. Specifically, after obtaining the activated semantic features from the autoencoder, each feature is paired with its textual explanation and predefined semantic attributes (as illustrated in Figure 5.1). These are then passed to GPT-3.5-turbo together with the sentence context in a structured prompt (see Table ??), which instructs the model to analyze the contextual meaning of the target word, compute the semantic alignment between each feature and the surrounding context, and return the features ranked by relevance. If the top-ranked feature does not correspond to the intended meaning or is inconsistent with the context, the target word is flagged as potentially ambiguous, triggering the GPT-4 clarification step.

Then, GPT-4 is prompted as a figurative language expert, tasked with disambiguating the meaning of a target word based on its sentence context and the semantic features extracted from the sparse autoencoder. The goal is to produce a concise clarification rather than rewriting the entire sentence, thus minimizing perturbation to the input while providing precise semantic guidance. The query format explicitly specifies the input sentence, the target word, and its top-k activated features, enabling fine-grained control over the interpretability process.

Prompt	You are a mathematical reasoning assistant tasked with disambiguating math problems. Given a math problem and a list of ambiguous symbols or expressions, rewrite the problem to clarify their meaning while making minimal changes to the original text. Instructions: 1. Preserve the structure and style of the original problem as much as possible. 2. Modify **only** the ambiguous symbols or expressions to make their meaning clear. 3. Do NOT solve the problem. 4. Output only the rewritten problem text, no additional commentary.
Query	**Original Problem**: {math_problem} **Ambiguous Symbols/Expressions**: {ambiguous_symbols_list}

Table 5.4: *Prompt and Query Format for LLMs in the Math question answering task pipeline, which is provided to GPT-4 for generating the rewritten problem.*

Prompt	You are an evaluator that determines whether two math problems are semantically equivalent. Instructions: 1. Compare the original problem and the rewritten problem. 2. Answer only with "Yes" if they express the same mathematical question (even if wording is different) or "No" if the meaning has changed. 3. Do not explain your answer.
Query	**Original Problem**: {original_problem} **Rewritten Problem**: {rewritten_problem}

Table 5.5: *Prompt and Query Format for LLMs in Math question answering task pipeline, which is used with Gemini 1.0-Pro to verify the semantic equivalence between the original and rewritten problems.*

Similarly, in the mathematical reasoning pipeline, GPT-4 is employed to rewrite mathematical problems in cases where ambiguous symbols or expressions have been identified through rule-based feature analysis. The prompt enforces minimal modifications, instructing the model to clarify only the specified symbols while preserving the original problem's structure and semantics. To further ensure that the meaning of the problem is preserved, a separate verification step is included: Gemini 1.0-Pro compares the original and rewritten problems and outputs a binary judgment of semantic equivalence. This additional step acts as a safeguard, preventing over-interpretation or unintended changes during the rewriting process.

This design facilitates replicability, as future researchers can apply the same prompts and query formats to reproduce or extend our results.

5.2.4 Replaceability of Reformulation

The core objective of this study is to explore an adaptive interpretability method based on LLM-internal features, rather than merely relying on a more powerful model for input reformulation. Our approach does not simply modify inputs but instead leverages LLM-internal feature analysis to predict ambiguity and employs external knowledge to enhance interpretability. Fully relying on GPT-4 for reformulation would effectively outsource the input modality to a stronger model, bypassing the opportunity for LLMs to develop their own reasoning and generalization capabilities. This would not only limit the applicability of our method but also make it difficult to scale to computationally constrained environments. Additionally, our framework is designed as a generalizable interpretability enhancement pipeline, ensuring adaptability to a broader LLM ecosystem rather than restricting it to GPT-4 as an auxiliary tool.

Although this study employs GPT-4 for generating supplementary information, the approach is not inherently dependent on GPT-4 as the only option. The reformulation mechanism is fundamentally driven by LLMs-autoencoder internal feature extraction and analysis. GPT-4 is only triggered when LLMs-autoencoder predicts a high probability of ambiguity, providing necessary supplementary information rather than globally rewriting the input. Furthermore, this approach can be extended to

Math Query	**Response**	**Label**
If $\lvert 4x+2\rvert = 10$ and $x < 0$, what is the value of x?	Let's assume that x is a positive integer. Then, we can write the equation: $4x + 2 = 10$ Subtracting 2 from both sides, we get: $4x = 8$ Dividing both sides by 4, we get: $x = 2$ Therefore, the value of x is **2**.	✗
If the absolute value of 4x+2 equal to 10 and x is less than 0, what is the value of x?	Let's assume that x is less than 0. According to the given information, the absolute value of 4x+2 is equal to 10. So, we can write the equation as: —4x+2— = 10 To solve this equation, we can consider two cases:Case 1: 4x+2 = 10 Subtracting 2 from both sides of the equation, we get: 4x = 8 Dividing both sides of the equation by 4, we get: x = 2. Case 2: 4x+2 = -10 Subtracting 2 from both sides of the equation, we get: 4x = -12 Dividing both sides of the equation by 4, we get: x = -3 Since x is less than 0, the only valid solution is x = -3. Therefore, the value of x is **-3**.	✓
Metaphor Query	**Response**	**Label**
The champagne flowed at the wedding. Is the target word 'flowed' a metaphorical or literal expression?	The phrase "The champagne flowed at the wedding" is a **literal** expression. In this context, the word 'flowed' describes the actual movement of champagne being poured and spilling freely as would be expected of sparkling wine.	✗
The champagne flowed at the wedding. 'flowed' implies a free and plentiful availability. Is the target word 'flowed' a metaphorical or literal expression?	In the context of the sentence, "The champagne flowed at the wedding," the term 'flowed' is being used in a **metaphorical** sense. Literally, 'flowed' means to move smoothly and continuously like water or another liquid. Here, it isn't used to describe the actual movement of champagne but rather the abundance and continuous presence of champagne at the wedding, almost like a flowing water current.	✓

Table 5.6: *The comparison of the pair of Query & Response between the original question and the reformed question. The responses are generated by Phi-3, and the reformulated query is enhanced by adopting the LLMs-Autoencoder pipeline.*

other LLMs for knowledge supplementation or adapted using fine-tuned instruction models (e.g., T5, LLaMA) or rule-based methods to achieve similar semantic enhancement, improving scalability and computational efficiency in resource-constrained environments.

5.3 Experiments

5.3.1 Dataset

Training data The training data used for the sparse autoencoder in dictionary learning is proportionally sampled from the open-source LLMs pre-training dataset, ***RedPajama***,[1] ensuring that the capabilities of models across various aspects are preserved. To improve the model's understanding of downstream tasks, we also incorporated an open-source mathematics dataset, sampling approximately 200M tokens. ***OpenMathInstruct*** (Toshniwal et al. 2024) is a math instruction tuning dataset generated using permissively licensed Mixtral-8x7B models.

Metaphor test data We evaluate our approach on the MOH and TroFi datasets (see detailed descriptions in §2.2.3). These benchmarks cover both short (MOH) and long (TroFi) sentences, with each annotated for metaphorical and literal verb usage.

Mathematica test data (*MATH*) (Hendrycks et al. 2021) is an open-source math dataset of 12,500 challenging competition mathematics problems. Each problem in MATH has a full step-by-step solution to evaluate the reasoning capabilities of models in the domain of mathematics.

5.3.2 Large Language Models

We evaluate the effectiveness of our method using four state-of-the-art LLMs, each having around 7 billion parameters: Llama 3 (3.1-8B-Instruct) (Dubey et al. 2024), Mistral (7B-Instruct) (Jiang et al. 2023), Gemma (7b-it) (Mesnard et al. 2024), and Phi-3 (Small-8K-Instruct) (Abdin et al. 2024). These models serve as the backbone of our experimental analysis, where we apply our feature extraction and metaphor detection enhancement techniques to assess their ability to interpret and resolve ambiguous metaphorical expressions.

Source	**Tokens (Billion)**
Common Crawl	8.75
C4	1.75
GitHub	0.60
ArXiv	0.30
Books	0.25
Wikipedia	0.25
StackExchange	0.20
OpenMathInstruct	0.20
Total	**12.3**

Table 5.7: *Overview of the data sources used for training, including web crawls (Common Crawl, C4), code repositories (GitHub), scientific articles (arXiv), encyclopedic content (Wikipedia), question-answering platforms (StackExchange), and a specialized dataset for mathematical instruction (OpenMathInstruct).*

5.3.3 Data Mixing

To ensure that our model maintains strong versatility and performs well in a range of applications, rather than being over-specialized to a single task, we adopted a data mixing training approach. The distribution of data significantly influences the learned weights in the auto-encoder (Elhage et al. 2022), particularly in how different features are extracted. To develop a more comprehensive and task-agnostic feature dictionary, we generated the training data by sampling from different LLM pre-training datasets and incorporated portions of downstream task datasets. This approach ensures the learned dictionary is more versatile and applicable to diverse inputs and tasks. The specific data ratios used in training are shown in Table 5.7.

5.4 Results

5.4.1 Metaphor Detection

Unlike the ambiguity found in mathematical symbols, linguistic ambiguity has long been a significant focus in NLP. Our approach directly extracts internal features from the model, allowing for a more explicit and detailed analysis of how the model understands metaphorical expressions.

[1]Dataset available at https://huggingface.co/datasets/togethercomputer/RedPajama-Data-1T.

Among the baselines, RoPPT represents our own prior work introduced in Chapter 3, which incorporates syntactic structures to denoise and improve metaphor detection. MelBERT (Choi et al. 2021) and MrBERT (Song et al. 2021) are earlier transformer-based models, that MelBERT models metaphor via contextual–literal contrasts (MIP, SPV), while MrBERT incorporates verb–argument structures. AdMul (Zhang & Liu 2023) uses adaptive multi-task learning with multiple external knowledge sources, but its results are not fully reproducible and are shown here for reference only.

The results in Table 5.8 reveal insights into the effectiveness of our method in improving metaphor detection with various LLMs. Notably, models like Llama 3 and Mistral demonstrated substantial gains, with Llama 3 improving by 4.3% on MOH-X and 4% on Tro-Fi, and Mistral seeing similar increases across both datasets. These improvements suggest that our approach enhances the models' ability to handle the inherent ambiguity in metaphorical expressions. By addressing this challenge, our method enables more accurate metaphor recognition, showing that it outperforms previous approaches in both generalization and handling linguistic complexity.

Additionally, investigation into the understanding of the LLMs via their activations reveals the potential for ambiguity in existing metaphor detection datasets. For instance, the TroFi (Birke & Sarkar 2006) dataset states that in "an aerodynamic shape that reduces *drag* 20 % to 25 %" and "Their melanin - producing system keeps the skin constantly *filled* with the dark pigment", the italicized words ("drag" and "filled") are being used in a metaphorical sense, whilst we observe highest activation in features relating to their literal interpretations. In such cases, these examples do not align with lay understanding of what constitutes metaphor (which are usually more literary), with both being very close to their literal interpretations.

We assessed the significance of accuracy improvements between the original and enhanced models using paired t-tests across all metaphor dateset. All p-values are below 0.05, confirming statistically significant gains for Llama 3, Mistral, Gemma, and Phi-3. McNemar's tests on prediction outputs further support these results with $p < 0.001$ for all models. Similar trends were observed on the following mathematical question answering task.

Method	**MOH-X**	**Tro-Fi**
MelBERT	79.2	62.0
MrBERT	79.8	62.7
RoPPT	80.1	63.3
AdMul	83.9	63.3
Llama 3	80.1	61.5
Llama 3 Enhanced	**84.4**	**65.5**
Mistral	78.5	60.8
Mistral Enhanced	**83.6**	**63.1**
Gemma	78.9	60.9
Gemma Enhanced	**82.9**	**63.8**
Phi-3	76.4	59.3
Phi-3 Enhanced	**80.7**	**62.5**

Table 5.8: *Comparison of metaphor detection accuracy. We include task-specific models and general-purpose LLMs before and after enhancement using internal feature extraction techniques.* ***Bold*** *represents the highest-scoring version for each model and dataset. In all cases, our automatically enhanced prompts perform the best, with an average absolute increase of 3.76% (a relative increase of 5.38%).*

5.4.2 Mathematics Question-Answering

We confirmed that applying detection and problem-enhancement methods leads to consistent performance improvements across multiple models and mathematical domains. We use the math equivalence python script of MATH dataset(Hendrycks et al. 2021) provided on their official github [2], which extracts and compares only the final boxed answer for correctness. As shown in Table 5.9, all models experienced significant accuracy gains after using the reformed query method, which integrates ambiguity detection and rephrasing techniques.

The results show significant accuracy improvements, shown in Table 5.9, particularly in domains requiring complex reasoning. Llama 3 saw the most substantial increase in counting and probability, while Mistral and Phi made notable gains in

[2]The official code and data of MATH dataset could be downloaded via this link: https://github.com/hendrycks/math.

prealgebra. Gemma improved in both intermediate algebra and prealgebra. These improvements highlight the effectiveness of rephrasing in enhancing the models' ability to handle diverse mathematical problems more effectively. To evaluate the statistical significance of the observed performance gains, we conducted McNemar's tests at $\alpha = 0.05$ significance level. The paired tests were performed both on the overall MATH dataset (12,500 problems) and individually on each subdomain (e.g., Algebra, Counting Probability, Number Theory, Precalculus, Geometry). The results confirm that the improvements achieved with the reformed query method are statistically significant across all models, both in the aggregate and within each mathematical subfield. These findings provide strong evidence that the ambiguity detection and prompt rephrasing techniques significantly enhance model performance in mathematical question-answering tasks.

In summary, across all models and mathematical domains, the reformed query method consistently yielded higher accuracy, confirming the effectiveness of detection and rephrasing techniques in improving the interpretability and performance of mathematical question answering.

Model	Intermediate Algebra	Counting/Probability	Precalculus	Number Theory	Algebra	Prealgebra	Geometry	Total
Llama 3 Original	27.6	23.9	26.7	23.1	34.0	37.5	34.9	30.6
Llama 3 Enhanced	**40.3**	**47.3**	**39.1**	**46.7**	**53.1**	**63.0**	**41.1**	**48.6**
Mistral Original	25.8	26.1	22.4	24.4	29.8	36.6	27.7	28.3
Mistral Enhanced	**36.5**	**39.9**	**37.9**	**36.6**	**41.5**	**46.8**	**38.6**	**41.2**
Gemma Original	29.6	23.1	23.7	27.2	27.8	35.2	27.5	27.9
Gemma Enhanced	**40.9**	**33.8**	**35.5**	**35.5**	**35.1**	**46.4**	**39.1**	**38.6**
Phi-3 Original	16.9	18.1	21.3	17.3	25.2	30.1	28.7	22.9
Phi-3 Enhanced	**27.2**	**29.3**	**30.5**	**26.7**	**35.0**	**46.1**	**31.4**	**33.2**

Table 5.9: *Comparison of answering accuracy across four models (Llama-3, Mistral, Gemma, and Phi-3) in different mathematical domains, using original and reformed query methods.* ***Bold*** *represents the highest-scoring version for each model and question type. In all cases, our automatically enhanced prompts perform the best, with an average absolute increase of 12.52% (a relative increase of 47.78%).*

5.4.3 Math Error Analysis

While analyzing a collection of mathematical errors, we categorized mathematical symbols into three distinct types: (1) **Functions**: Include functions such as *cos*, *sin*, and *sqrt*, among others. (2) **Operators**: Consisting of special characters or algebraic symbols, this category includes symbols like +, -, or algebraic variables such as x and y. These symbols are often ambiguous due to their polysemantic nature and different usages outside of mathematical contexts (e.g., the less-than sign ”$<$” or absolute value symbols in Table 5.1). (3) **Numbers**: Referring to common numerical digits.

Among these types, numbers trigger the least ambiguity during mathematical computations. However, ambiguity is more prevalent in the other two categories, albeit for different reasons. **Operators** tend to create confusion because of their ubiquity in non-mathematical contexts. For example, as previously discussed, symbols like the less-than sign and the absolute value symbol can be misinterpreted when applied outside of a purely mathematical framework. Variables like x and y are often included in this category due to their frequent use across various disciplines. **Functions** present a more complex challenge, sometimes being mistakenly identified as code-related functions rather than mathematical functions, complicating interpretation. Furthermore, issues such as tokenization add to the difficulty. For instance, the LaTeX function \dfrac (which denotes fractions in display style) can be broken down into tokens like ’\’, ’d’, and ’frac’, making it even harder to analyze correctly.

5.5 Conclusion

We investigated the extent to which the decomposition of traditionally black-box LLMs can aid in increasing their interpretability and lead to performance improvements in downstream tasks such as mathematical reasoning and metaphor detection. We observe improvements from integrating our dictionary learning approach to extract monosemantic features, allowing the identification of which features are activated by a task, to then rewrite the prompt with additional task information and improve performance. We were further able to identify the key misinterpretations being made by LLMs on these tasks via feature activations, in turn identifying the challenges posed by mathematical notation and ambiguous metaphorical usage.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

Metaphors are fundamental to human language, cognition, and communication, serving as both a conceptual framework and a linguistic tool for expressing complex ideas. This thesis has explored the intricate relationship between metaphors and natural language processing (NLP), delving into the challenges of metaphor detection, interpretation, and translation. By bridging linguistic theories with computational approaches, this thesis has developed and evaluated computational models that reveal the limitations of the current state in processing metaphors, and the difficulties they pose for artificial intelligence systems.

The discussions presented throughout this thesis highlight the importance of metaphor processing in NLP applications such as sentiment analysis, machine translation, and creative language generation. Traditional models struggle with figurative expressions due to their reliance on literal semantic mappings, while recent advancements in deep learning and large language models (LLMs) have opened new possibilities for handling metaphorical language more effectively. However, as shown in our exploration of metaphor understanding in LLMs, challenges such as contextual ambiguity, cultural variance, and the interpretability of AI-generated metaphors remain significant hurdles.

A key focus of this thesis has been the cross-linguistic perspectives on metaphors, emphasizing the need for culturally adaptive NLP systems that can recognize and accurately translate metaphorical expressions across different languages. Through the development of metaphor-specific evaluation frameworks and the introduction of datasets for metaphor translation in Chapter 4, we have made a contribution for future research in computational metaphor processing. These contributions aim to improve not only the accuracy of NLP models but also their ability to capture the richness and subtlety of human expression.

Our work also advances the interpretability of large language models (LLMs) in metaphor processing by addressing their inherent black-box nature through feature decomposition techniques. While LLMs have demonstrated remarkable abilities in metaphor detection and generation, their lack of transparency makes it difficult to pinpoint how they process figurative language, leading to potential errors and biases. By integrating dictionary learning to extract monosemantic features, we have significantly improved LLM interpretability, allowing us to identify which specific features are activated when processing metaphors. This insight not only enhances our understanding of how LLMs process metaphorical expressions but also enables us to refine model inputs through prompt rewriting, ultimately improving metaphor detection performance. Furthermore, our approach reveals key misinterpretations made by LLMs, shedding light on the challenges posed by ambiguous metaphorical usage and context-dependent figurative meaning. By bridging interpretability research with metaphor processing, our contributions provide a solid contribution for more transparent, controllable, and linguistically informed LLMs, fostering better alignment with human cognitive and communicative frameworks.

6.2 Limitations of Current Work

Although this thesis makes several contributions to metaphor detection, interpretation, and translation, there are still notable limitations that need to be acknowledged.

First, most of the current work focuses on single-word metaphor detection, particularly verbs, and does not sufficiently address more complex multi-word metaphorical expressions. In real-world corpora, metaphors often occur as phrases or even entire sentences, such as “kick the bucket” or “spill the beans.” Identifying and interpreting such multi-word metaphors remains an open challenge.

Second, the task setup used in this thesis assumes that the target word is given, meaning the input explicitly specifies the word to be judged. However, in real-world applications, models must often identify metaphorical expressions in raw text without knowing the target word in advance. This more realistic setting is much more challenging because the model must both locate potential targets and decide whether they are used metaphorically.

In addition, this research is primarily based on English corpora and English–Chinese-Italian translation pairs, with limited coverage of other languages and cultures. Since the use of metaphors varies significantly across languages and cultures, further work is needed to evaluate the cross-lingual robustness and generalizability of the proposed methods.

Finally, while this thesis introduces interpretability methods such as sparse autoencoders, there has not yet been large-scale human validation of the interpretability of the learned features, and the clarification-based reformulation strategy has only been evaluated in controlled experiments rather than on large, naturally distributed datasets.

6.3 Future Work

Our work has made progress in metaphor detection, translation, and interpretation, but as noted above, there are still many limitations. Therefore, we propose several specific directions for future research, aiming to address these limitations and further advance the state of metaphor processing.

First, for the metaphor detection models developed in Chapter 3, one natural extension would be to combine the different approaches—such as context denoising, frame-based modeling, and explicit basic-meaning modeling—into an ensemble system. Since each method captures different aspects of metaphor, ensembling could lead to more robust and accurate detection.

Second, the corpus introduced in Chapter 4 could be expanded and enriched. In particular, adding more languages (e.g., Spanish, Arabic) would allow us to study how metaphor translation challenges vary across cultures. Beyond single-word verbs, future versions of the corpus should also include multi-word metaphors, which represent a much more difficult and realistic setting. Additional annotation metrics, such as metaphor categories, could also provide richer resources for downstream tasks.

Third, the interpretability pipeline presented in Chapter 5 could be further evaluated and extended. A clear next step is to run human evaluations of the dictionary features discovered by the sparse autoencoder, in order to assess how understandable and useful they are for human annotators. Another direction is to move from the current setup, where the target word is given, to the harder problem of identifying metaphors in a sentence, where the system must find and interpret metaphorical expressions without prior hints.

Finally, an important long-term goal is to make metaphor processing more inclusive for low-resource languages. By combining transfer learning with the interpretability-guided pipeline, we could explore few-shot or zero-shot setups, enabling models to handle metaphorical language even when little or no training data is available.

Taken together, these directions would help refine the proposed methods, broaden the coverage of the resources introduced here, and test the models in more realistic scenarios, making them better aligned with the complexities of real-world metaphor use.

Bibliography

Abd Yusof, N. F., Lin, C. & Guerin, F. (2017), Analysing the causes of depressed mood from depression vulnerable individuals, *in* 'Proceedings of the International Workshop on Digital Disease Detection using Social Media 2017 (DDDSM-2017)', pp. 9–17.

Abdin, M., Jacobs, S. A., Awan, A. A., Aneja, J., Awadallah, A., Awadalla, H., Bach, N., Bahree, A., Bakhtiari, A., Behl, H. et al. (2024), 'Phi-3 technical report: A highly capable language model locally on your phone', *arXiv preprint arXiv:2404.14219* .

Abnar, S. & Zuidema, W. (2020), Quantifying attention flow in transformers, *in* D. Jurafsky, J. Chai, N. Schluter & J. Tetreault, eds, 'Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics', Association for Computational Linguistics, Online, pp. 4190–4197.
URL: *https://aclanthology.org/2020.acl-main.385*

Agirre, E. & Stevenson, M. (2007), 'Knowledge sources for wsd', *Word sense disambiguation: Algorithms and applications* **33**, 217.

Allen, J. (2003), 'Post-editing', *Benjamins Translation Library* **35**, 297–318.

Barnden, J. (2020), Metaphor thoughtfully, *in* 'Figurative meaning construction in thought and language', John Benjamins Publishing Company, pp. 13–44.

Barnden, J. (2021), Metaphor and irony: Messy when mixed, *in* 'Figurative Language–Intersubjectivity and Usage', John Benjamins, pp. 139–174.

Belinkov, Y. (2022), 'Probing classifiers: Promises, shortcomings, and advances', *Computational Linguistics* **48**(1), 207–219.
URL: *https://aclanthology.org/2022.cl-1.7*

Belinkov, Y. & Glass, J. (2019), 'Analysis methods in neural language processing: A survey', *Transactions of the Association for Computational Linguistics* **7**, 49–72.
URL: *https://aclanthology.org/Q19-1004*

Bergsma, S., Lin, D. & Goebel, R. (2008), Distributional identification of non-referential pronouns, *in* 'Proceedings of ACL-08: HLT', pp. 10–18.

Bills, S., Cammarata, N., Mossing, D., Tillman, H., Gao, L., Goh, G., Sutskever, I., Leike, J., Wu, J. & Saunders, W. (2023), 'Language models can explain neurons in language models', *OpenAI Blog* .
URL: *https://openaipublic.blob.core.windows.net/neuron-explainer/paper/index.html*

Birke, J. & Sarkar, A. (2006), A clustering approach for nearly unsupervised recognition of nonliteral language, *in* D. McCarthy & S. Wintner, eds, '11th Conference of the European Chapter of the Association for Computational Linguistics', Association for Computational Linguistics, Trento, Italy, pp. 329–336.
URL: *https://aclanthology.org/E06-1042*

Bizzoni, Y. & Lappin, S. (2018), Predicting human metaphor paraphrase judgments with deep neural networks, *in* 'Proceedings of the workshop on figurative language processing', pp. 45–55.

Brants, T. (2006), 'Web 1t 5-gram version 1', *Philadelphia Linguistic Data Consortium* .

Bricken, T., Templeton, A., Batson, J., Chen, B., Jermyn, A., Conerly, T., Turner, N., Anil, C., Denison, C., Askell, A. et al. (2023), 'Towards monosemanticity: Decomposing language models with dictionary learning', *Transformer Circuits Thread* **2**.

Brinton, L. J. & Brinton, D. M. (2010), 'The linguistic structure of modern english'.

Broadwell, G. A., Boz, U., Cases, I., Strzalkowski, T., Feldman, L., Taylor, S., Shaikh, S., Liu, T., Cho, K. & Webb, N. (2013), Using imageability and topic chaining to locate metaphors in linguistic corpora, *in* 'International Conference on Social Computing, Behavioral-Cultural Modeling, and Prediction', Springer, pp. 102–110.

Bulat, L., Clark, S. & Shutova, E. (2017), Modelling metaphor with attribute-based semantics, *in* 'Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers', pp. 523–528.

Carroll, J. B. (1966), 'An experiment in evaluating the quality of translations.', *Mech. Transl. Comput. Linguistics* **9**(3-4), 55–66.

Chen, X., Hai, Z., Wang, S., Li, D., Wang, C. & Luan, H. (2021), 'Metaphor identification: A contextual inconsistency based neural sequence labeling approach', *Neurocomputing* **428**, 268–279.

Choi, M., Lee, S., Choi, E., Park, H., Lee, J., Lee, D. & Lee, J. (2021), MelBERT: Metaphor detection via contextualized late interaction using metaphorical identification theories, *in* K. Toutanova, A. Rumshisky, L. Zettlemoyer, D. Hakkani-Tur, I. Beltagy, S. Bethard, R. Cotterell, T. Chakraborty & Y. Zhou, eds, 'Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies', Association for Computational Linguistics, Online, pp. 1763–1773.
URL: *https://aclanthology.org/2021.naacl-main.141*

Church, K. W. & Hovy, E. H. (1993), 'Good applications for crummy machine translation', *Machine Translation* **8**(4), 239–258.

Conneau, A., Kruszewski, G., Lample, G., Barrault, L. & Baroni, M. (2018), What you can cram into a single $&!#* vector: Probing sentence embeddings for linguistic properties, *in* I. Gurevych & Y. Miyao, eds, 'Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)', Association for Computational Linguistics, Melbourne, Australia,

pp. 2126–2136.
URL: *https://aclanthology.org/P18-1198*

Cunningham, H., Ewart, A., Riggs, L., Huben, R. & Sharkey, L. (2023), 'Sparse autoencoders find highly interpretable features in language models', *arXiv preprint arXiv:2309.08600* .

Devlin, J., Chang, M.-W., Lee, K. & Toutanova, K. (2019), BERT: Pre-training of deep bidirectional transformers for language understanding, *in* J. Burstein, C. Doran & T. Solorio, eds, '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)', Association for Computational Linguistics, Minneapolis, Minnesota, pp. 4171–4186.
URL: *https://aclanthology.org/N19-1423*

Dietterich, T. G. (1998), 'Approximate statistical tests for comparing supervised classification learning algorithms', *Neural computation* **10**(7), 1895–1923.

Do Dinh, E.-L. & Gurevych, I. (2016), Token-level metaphor detection using neural networks, *in* 'Proceedings of the Fourth Workshop on Metaphor in NLP', pp. 28–33.

Doshi-Velez, F. & Kim, B. (2017), 'Towards a rigorous science of interpretable machine learning', *arXiv preprint arXiv:1702.08608* .

Doyon, J., Taylor, K. B. & White, J. S. (1999), Task-based evaluation for machine translation, *in* 'Proceedings of Machine Translation Summit VII', pp. 574–578.

Dozat, T. & Manning, C. D. (2016), 'Deep biaffine attention for neural dependency parsing', *arXiv preprint arXiv:1611.01734* .

Dubey, A., Jauhri, A., Pandey, A., Kadian, A., Al-Dahle, A. et al. (2024), 'The llama 3 herd of models'.
URL: *https://arxiv.org/abs/2407.21783*

Elhage, N., Hume, T., Olsson, C., Schiefer, N., Henighan, T., Kravec, S., Hatfield-Dodds, Z., Lasenby, R., Drain, D., Chen, C. et al. (2022), 'Toy models of superposition', *arXiv preprint arXiv:2209.10652* .

Faruqui, M., Tsvetkov, Y., Yogatama, D., Dyer, C. & Smith, N. A. (2015), Sparse overcomplete word vector representations, *in* C. Zong & M. Strube, eds, 'Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)', Association for Computational Linguistics, Beijing, China, pp. 1491–1500.
URL: *https://aclanthology.org/P15-1144*

Gallant, S. I. (1991), 'A practical approach for representing context and for performing word sense disambiguation using neural networks', *Neural Computation* **3**(3), 293–309.

Gao, G., Choi, E., Choi, Y. & Zettlemoyer, L. (2018), 'Neural metaphor detection in context', *arXiv preprint arXiv:1808.09653* .

Ge, M., Mao, R. & Cambria, E. (2022), Explainable metaphor identification inspired by conceptual metaphor theory, *in* 'Proceedings of the AAAI conference on artificial intelligence', Vol. 36, pp. 10681–10689.

Ghosh, A., Li, G., Veale, T., Rosso, P., Shutova, E., Barnden, J. & Reyes, A. (2015), Semeval-2015 task 11: Sentiment analysis of figurative language in twitter, *in* 'Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015)', pp. 470–478.

Gong, H., Gupta, K., Jain, A. & Bhat, S. (2020), Illinimet: Illinois system for metaphor detection with contextual and linguistic information, *in* 'Proceedings of the second workshop on figurative language processing', pp. 146–153.

Gupta, P. & Schütze, H. (2018), LISA: Explaining recurrent neural network judgments via layer-wIse semantic accumulation and example to pattern transformation, *in* T. Linzen, G. Chrupała & A. Alishahi, eds, 'Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP', Association for Computational Linguistics, Brussels, Belgium, pp. 154–164.
URL: *https://aclanthology.org/W18-5418*

Gutiérrez, E. D., Cecchi, G., Corcoran, C. & Corlett, P. (2017), Using automated metaphor identification to aid in detection and prediction of first-episode schizophrenia, *in* 'Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing', Association for Computational Linguistics, Copenhagen, Denmark, pp. 2923–2930.
URL: *https://aclanthology.org/D17-1316*

Hartley, R. T. & Barnden, J. A. (1997), 'Semantic networks: visualizations of knowledge', *Trends in Cognitive Sciences* **1**(5), 169–175.

Hendrycks, D., Burns, C., Kadavath, S., Arora, A., Basart, S., Tang, E., Song, D. & Steinhardt, J. (2021), 'Measuring mathematical problem solving with the math dataset', *NeurIPS* .
URL: *https://datasets-benchmarks-proceedings.neurips.cc/paper/2021/file/be83ab3ecd0db773eb2dc1b0a17836a1-Paper-round2.pdf*

Honnibal, M. & Montani, I. (2017), spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear.

Jiang, A. Q., Sablayrolles, A., Mensch, A., Bamford, C., Chaplot, D. S., Casas, D. d. l., Bressand, F., Lengyel, G., Lample, G., Saulnier, L. et al. (2023), 'Mistral 7b', *arXiv preprint arXiv:2310.06825* .

Jiang, T. & Riloff, E. (2021), Learning prototypical functions for physical artifacts, *in* C. Zong, F. Xia, W. Li & R. Navigli, eds, '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)', Association for Computational Linguistics, Online, pp. 6941–6951.
URL: *https://aclanthology.org/2021.acl-long.540/*

Jurafsky, D. & Martin, J. H. (2025), *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition with Language Models*, 3rd edn. Online manuscript released August

24, 2025.
URL: *https://web.stanford.edu/ jurafsky/slp3/*

Karpathy, A., Johnson, J. & Fei-Fei, L. (2015), 'Visualizing and understanding recurrent networks', *arXiv preprint arXiv:1506.02078* .

Kehat, G. & Pustejovsky, J. (2021), Neural metaphor detection with visibility embeddings, *in* 'Proceedings of* SEM 2021: The Tenth Joint Conference on Lexical and Computational Semantics', pp. 222–228.

Krishnakumaran, S. & Zhu, X. (2007), Hunting elusive metaphors using lexical resources., *in* 'Proceedings of the Workshop on Computational approaches to Figurative Language', pp. 13–20.

Lakoff, G. (1993), 'The contemporary theory of metaphor'.

Lakoff, G. & Johnson, M. (1980), 'Metaphors we live by: University of chicago press', *Chicago, IL* .

Le, D., Thai, M. & Nguyen, T. (2020), Multi-task learning for metaphor detection with graph convolutional neural networks and word sense disambiguation, *in* 'Proceedings of the AAAI conference on artificial intelligence', Vol. 34, pp. 8139–8146.

Lee, H., Battle, A., Raina, R. & Ng, A. (2006), Efficient sparse coding algorithms, *in* B. Schölkopf, J. Platt & T. Hoffman, eds, 'Advances in Neural Information Processing Systems', Vol. 19, MIT Press.

Lee, H., Ekanadham, C. & Ng, A. (2007), 'Sparse deep belief net model for visual area v2', *Advances in neural information processing systems* **20**.

Leong, C. W., Klebanov, B. B., Hamill, C., Stemle, E., Ubale, R. & Chen, X. (2020), A report on the 2020 vua and toefl metaphor detection shared task, *in* 'Proceedings of the Second Workshop on Figurative Language Processing', pp. 18–29.

Leong, C. W., Klebanov, B. B. & Shutova, E. (2018), A report on the 2018 vua metaphor detection shared task, *in* 'Proceedings of the Workshop on Figurative Language Processing', pp. 56–66.

Likert, R. (1932), 'A technique for the measurement of attitudes.', *Archives of psychology* .

Lin, C.-Y. (2004), Rouge: A package for automatic evaluation of summaries, *in* 'Text summarization branches out', pp. 74–81.

Liu, N. F., Levy, O., Schwartz, R., Tan, C. & Smith, N. A. (2018), LSTMs exploit linguistic attributes of data, *in* I. Augenstein, K. Cao, H. He, F. Hill, S. Gella, J. Kiros, H. Mei & D. Misra, eds, 'Proceedings of the Third Workshop on Representation Learning for NLP', Association for Computational Linguistics, Melbourne, Australia, pp. 180–186.
URL: *https://aclanthology.org/W18-3024*

Liu, Y. (2019), 'Roberta: A robustly optimized bert pretraining approach', *arXiv preprint arXiv:1907.11692* **364**.

Maladry, A., Lefever, E., Van Hee, C. & Hoste, V. (2023), A fine line between irony and sincerity: Identifying bias in transformer models for irony detection, *in* J. Barnes, O. De Clercq & R. Klinger, eds, 'Proceedings of the 13th Workshop on Computational Approaches to Subjectivity, Sentiment, & Social Media Analysis', Association for Computational Linguistics, Toronto, Canada, pp. 315–324.
URL: *https://aclanthology.org/2023.wassa-1.28*

Mao, R. & Li, X. (2021), Bridging towers of multi-task learning with a gating mechanism for aspect-based sentiment analysis and sequential metaphor identification, *in* 'Proceedings of the AAAI conference on artificial intelligence', Vol. 35, pp. 13534–13542.

Mao, R., Li, X., Ge, M. & Cambria, E. (2022), 'Metapro: A computational metaphor processing model for text pre-processing', *Information Fusion* **86**, 30–43.

Mao, R., Lin, C. & Guerin, F. (2018), Word embedding and wordnet based metaphor identification and interpretation, *in* 'Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)', pp. 1222–1231.

Mao, R., Lin, C. & Guerin, F. (2019), End-to-end sequential metaphor identification inspired by linguistic theories, *in* 'Proceedings of the 57th annual meeting of the association for computational linguistics', pp. 3888–3898.

Martin, J. (1990), 'A computational model of metaphor interpretation', *San Diego, CA* .

Mason, Z. J. (2004), 'Cormet: a computational, corpus-based conventional metaphor extraction system', *Computational linguistics* **30**(1), 23–44.

Mesnard, T., Hardin, C., Dadashi, R., Bhupatiraju, S., Pathak, S. et al. (2024), 'Gemma: Open models based on gemini research and technology'.
URL: *https://arxiv.org/abs/2403.08295*

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. & Dean, J. (2013), 'Distributed representations of words and phrases and their compositionality (2013)', *arXiv preprint arXiv:1310.4546* .

Miller, G. A. (1995), 'Wordnet: a lexical database for english', *Communications of the ACM* **38**(11), 39–41.

Miller, G. A. (1998), *WordNet: An electronic lexical database*, MIT press.

Miller, G. A., Chodorow, M., Landes, S., Leacock, C. & Thomas, R. G. (1994), Using a semantic concordance for sense identification, *in* 'Human Language Technology: Proceedings of a Workshop held at Plainsboro, New Jersey, March 8-11, 1994'.

Mohammad, S., Shutova, E. & Turney, P. (2016), Metaphor as a medium for emotion: An empirical study, *in* 'Proceedings of the Fifth Joint Conference on Lexical and Computational Semantics', pp. 23–33.

Mohler, M., Bracewell, D., Tomlinson, M. & Hinote, D. (2013), Semantic signatures for example-based linguistic metaphor detection, *in* 'Proceedings of the First Workshop on Metaphor in NLP', pp. 27–35.

Murdoch, W. J., Liu, P. J. & Yu, B. (2018), 'Beyond word importance: Contextual decomposition to extract interactions from lstms', *arXiv preprint arXiv:1801.05453* .

Nair, V. & Hinton, G. E. (2009), '3d object recognition with deep belief nets', *Advances in neural information processing systems* **22**.

Nakayama, H., Kubo, T., Kamura, J., Taniguchi, Y. & Liang, X. (2018), 'doccano: Text annotation tool for human'. Software available from https://github.com/doccano/doccano.
URL: *https://github.com/doccano/doccano*

Neumann, M., King, D., Beltagy, I. & Ammar, W. (2019), 'Scispacy: fast and robust models for biomedical natural language processing', *arXiv preprint arXiv:1902.07669* .

Olah, C., Cammarata, N., Schubert, L., Goh, G., Petrov, M. & Carter, S. (2020), 'Zoom in: An introduction to circuits', *Distill* **5**(3), e00024–001.

Olshausen, B. A. & Field, D. J. (1997), 'Sparse coding with an overcomplete basis set: a strategy employed by v1?', *Vision Res* **37**(23), 3311–3325.

Oncevay, A., Haddow, B. & Birch, A. (2020), 'Bridging linguistic typology and multilingual machine translation with multi-view language representations', *arXiv preprint arXiv:2004.14923* .

Osbeck, L. M., Nersessian, N. J., Malone, K. R. & Newstetter, W. C. (2010), *Science as psychology: Sense-making and identity in science practice*, Cambridge University Press.

Pantazopoulos, G., Suglia, A., Lemon, O. & Eshghi, A. (2024), Lost in space: Probing fine-grained spatial understanding in vision and language resamplers, *in* K. Duh,

H. Gomez & S. Bethard, eds, 'Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)', Association for Computational Linguistics, Mexico City, Mexico, pp. 540–549.
URL: *https://aclanthology.org/2024.naacl-short.45*

Papineni, K., Roukos, S., Ward, T. & Zhu, W.-J. (2002), Bleu: a method for automatic evaluation of machine translation, *in* 'Proceedings of the 40th annual meeting of the Association for Computational Linguistics', pp. 311–318.

Peng, H., Thomson, S., Swayamdipta, S. & Smith, N. A. (2018), Learning joint semantic parsers from disjoint data, *in* M. Walker, H. Ji & A. Stent, eds, 'Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)', Association for Computational Linguistics, New Orleans, Louisiana, pp. 1492–1502.
URL: *https://aclanthology.org/N18-1135/*

Pennington, J., Socher, R. & Manning, C. D. (2014), Glove: Global vectors for word representation, *in* 'Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)', pp. 1532–1543.

Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K. & Zettlemoyer, L. (2018), 'Deep contextualized word representations', *arXiv preprint arXiv:1802.05365* .

Pragglejaz (2007), 'Mip: A method for identifying metaphorically used words in discourse', *Metaphor and symbol* **22**(1), 1–39.

Qian, P., Qiu, X. & Huang, X. (2016), Analyzing linguistic knowledge in sequential model of sentence, *in* J. Su, K. Duh & X. Carreras, eds, 'Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing', Association for Computational Linguistics, Austin, Texas, pp. 826–835.
URL: *https://aclanthology.org/D16-1079*

Rai, S., Chakraverty, S., Tayal, D. K., Sharma, D. & Garg, A. (2019), 'Understanding metaphors using emotions', *New Generation Computing* **37**, 5–27.

Rainio, O., Teuho, J. & Klén, R. (2024), 'Evaluation metrics and statistical tests for machine learning', *Scientific Reports* **14**(1), 6086.

Ratinov, L. & Roth, D. (2009), Design challenges and misconceptions in named entity recognition, *in* 'Proceedings of the thirteenth conference on computational natural language learning (CoNLL-2009)', pp. 147–155.

Ritter, A., Etzioni, O. et al. (2010), A latent dirichlet allocation method for selectional preferences, *in* 'Proceedings of the 48th annual meeting of the association for computational linguistics', pp. 424–434.

Rohanian, O., Rei, M., Taslimipoor, S. & Ha, L. (2020), Verbal multiword expressions for identification of metaphor, ACL.

Ruppenhofer, J., Baker, C. F. & Fillmore, C. J. (2002), The framenet database and software tools, *in* 'Proceedings of the Tenth Euralex International Congress. Copenhagen, Denmark', Vol. 1, pp. 371–375.

Senez, D. (1998), Post-editing service for machine translation users at the european commission, *in* 'Proceedings of Translating and the Computer 20'.

Shutova, E. (2010), Automatic metaphor interpretation as a paraphrasing task, *in* 'Human language technologies: the 2010 annual conference of the North American chapter of the association for computational linguistics', pp. 1029–1037.

Shutova, E. (2015), 'Design and evaluation of metaphor processing systems', *Computational Linguistics* **41**(4), 579–623.

Shutova, E., Kiela, D. & Maillard, J. (2016), Black holes and white rabbits: Metaphor identification with visual features, *in* 'Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: Human language technologies', pp. 160–170.

Shutova, E., Sun, L. & Korhonen, A. (2010), Metaphor identification using verb and noun clustering, *in* C.-R. Huang & D. Jurafsky, eds, 'Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)', Coling 2010 Organizing Committee, Beijing, China, pp. 1002–1010.
URL: *https://aclanthology.org/C10-1113*

Somers, H. (2003), 'Computers and translation', *Computers and Translation* pp. 1–365.

Song, L., Cui, Y., Luo, A., Lecue, F. & Li, I. (2024), Better explain transformers by illuminating important information, *in* Y. Graham & M. Purver, eds, 'Findings of the Association for Computational Linguistics: EACL 2024', Association for Computational Linguistics, St. Julian's, Malta, pp. 2048–2062.
URL: *https://aclanthology.org/2024.findings-eacl.138*

Song, W., Guo, J., Fu, R., Liu, T. & Liu, L. (2020), 'A knowledge graph embedding approach for metaphor processing', *IEEE/ACM Transactions on Audio, Speech, and Language Processing* **29**, 406–420.

Song, W., Zhou, S., Fu, R., Liu, T. & Liu, L. (2021), Verb metaphor detection via contextual relation learning, *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)', pp. 4240–4251.

Song, Y., Zhao, J. & Specia, L. (2021a), Sentsim: Crosslingual semantic evaluation of machine translation, *in* 'Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies', pp. 3143–3156.

Srivatsa, K. & Kochmar, E. (2024), 'What makes math word problems challenging for llms?', *arXiv preprint arXiv:2403.11369* .

Steen, G. (2010), *A method for linguistic metaphor identification: From MIP to MIPVU*, Vol. 14, John Benjamins Publishing.

Steen, G., Dorst, L., Herrmann, J., Kaal, A., Krennmayr, T. & Pasma, T. (2010), *A method for linguistic metaphor identification: From MIP to MIPVU.*

Su, C., Fukumoto, F., Huang, X., Li, J., Wang, R. & Chen, Z. (2020), Deepmet: A reading comprehension paradigm for token-level metaphor detection, *in* 'Proceedings of the Second Workshop on Figurative Language Processing', pp. 30–39.

Su, C., Huang, S. & Chen, Y. (2015), Context-dependent metaphor interpretation based on semantic relatedness, *in* 'Natural Language Processing and Chinese Computing: 4th CCF Conference, NLPCC 2015, Nanchang, China, October 9-13, 2015, Proceedings 4', Springer, pp. 182–193.

Su, C., Huang, S. & Chen, Y. (2017), 'Automatic detection and interpretation of nominal metaphor based on the theory of meaning', *Neurocomputing* **219**, 300–311.

Su, C., Peng, Y., Huang, S. & Chen, Y. (2020), 'A metaphor comprehension method based on culture-related hierarchical semantic model', *Neural Processing Letters* **51**, 2807–2826.

Su, C., Wu, K. & Chen, Y. (2021), Enhanced metaphor detection via incorporation of external knowledge based on linguistic theories, *in* 'Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021', pp. 1280–1287.

Su, X., Li, R., Li, X., Pan, J. Z., Zhang, H., Chai, Q. & Han, X. (2021), A knowledge-guided framework for frame identification, *in* C. Zong, F. Xia, W. Li & R. Navigli, eds, '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)', Association for Computational Linguistics, Online, pp. 5230–5240.
URL: *https://aclanthology.org/2021.acl-long.407/*

Tileagă, C. (2013), *Political psychology: Critical perspectives*, Cambridge University Press.

Toshniwal, S., Moshkov, I., Narenthiran, S., Gitman, D., Jia, F. & Gitman, I.

(2024), 'Openmathinstruct-1: A 1.8 million math instruction tuning dataset', *arXiv preprint arXiv:2402.10176* .

Tsvetkov, Y., Boytsov, L., Gershman, A., Nyberg, E. & Dyer, C. (2014), Metaphor detection with cross-lingual model transfer, *in* 'Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)', pp. 248–258.

Tsvetkov, Y., Mukomel, E. & Gershman, A. (2013), Cross-lingual metaphor detection using common semantic features, *in* 'Proceedings of the First Workshop on Metaphor in NLP', pp. 45–51.

Turney, P., Neuman, Y., Assaf, D. & Cohen, Y. (2011), Literal and metaphorical sense identification through concrete and abstract context, *in* 'Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing', pp. 680–690.

Wan, M. & Xing, B. (2020), Modality enriched neural network for metaphor detection, *in* 'Proceedings of the 28th International Conference on Computational Linguistics', pp. 3036–3042.

Wan, M., Xing, B., Su, Q., Liu, P. & Huang, C.-R. (2020), Sensorimotor enhanced neural network for metaphor detection, *in* 'Proceedings of the 34th pacific Asia conference on language, information and computation', pp. 312–317.

White, J. S., O'Connell, T. A. & O'Mara, F. E. (1994), The arpa mt evaluation methodologies: evolution, lessons, and future approaches, *in* 'Proceedings of the First Conference of the Association for Machine Translation in the Americas'.

Wilks, Y. (1975), 'A preferential, pattern-seeking, semantics for natural language inference', *Artificial intelligence* **6**(1), 53–74.

Wilks, Y. (1978), 'Making preferences more active', *Artificial intelligence* **11**(3), 197–223.

Wu, C., Wu, F., Chen, Y., Wu, S., Yuan, Z. & Huang, Y. (2018), Neural metaphor detecting with cnn-lstm model, *in* 'Proceedings of the Workshop on Figurative Language Processing', pp. 110–114.

Yun, Z., Chen, Y., Olshausen, B. & LeCun, Y. (2021), Transformer visualization via dictionary learning: contextualized embedding as a linear superposition of transformer factors, *in* E. Agirre, M. Apidianaki & I. Vulić, eds, 'Proceedings of Deep Learning Inside Out (DeeLIO): The 2nd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures', Association for Computational Linguistics, Online, pp. 1–10.
URL: *https://aclanthology.org/2021.deelio-1.1*

Zhang, S. & Liu, Y. (2023), 'Adversarial multi-task learning for end-to-end metaphor detection', *arXiv preprint arXiv:2305.16638* .

Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q. & Artzi, Y. (2019), 'Bertscore: Evaluating text generation with bert', *arXiv preprint arXiv:1904.09675* .

Statement on the Use of Generative AI in Thesis Preparation

In accordance with the University of Sheffield's principles for the use of Generative AI (GenAI) in learning and teaching, I provide the following statement to acknowledge the ways in which GenAI tools were used during the preparation of this thesis.

Acknowledge

Generative artificial intelligence (AI) technology, specifically ChatGPT (OpenAI, GPT-4), was used during the preparation of this thesis. Its application was strictly limited to supporting tasks such as language polishing, proofreading, and resolving LaTeX typesetting issues.

It is important to emphasise that at no stage did the AI tool generate core academic content, conduct data analysis, formulate original arguments, or interpret findings. All critical thinking, scholarly analysis, and final authorship remain exclusively my own.

The use of generative AI in this capacity is permitted under University policy, provided it is transparently declared. This declaration aligns with the University's principles on the use of AI in learning and teaching, and guidance for its declaration in assessments:

University of Sheffield AI Principles:
https://www.sheffield.ac.uk/saas/elevate/ai-principles

Guidance on Declaring AI Use:
https://www.sheffield.ac.uk/study-skills/digital/generative-ai/assessment

Describe

The use of GenAI was strictly limited to language-related tasks and followed the ethical, academic integrity, and data protection principles outlined by the University. Specifically, I used the tool for:

- Reviewing the thesis for grammar and typographical errors.

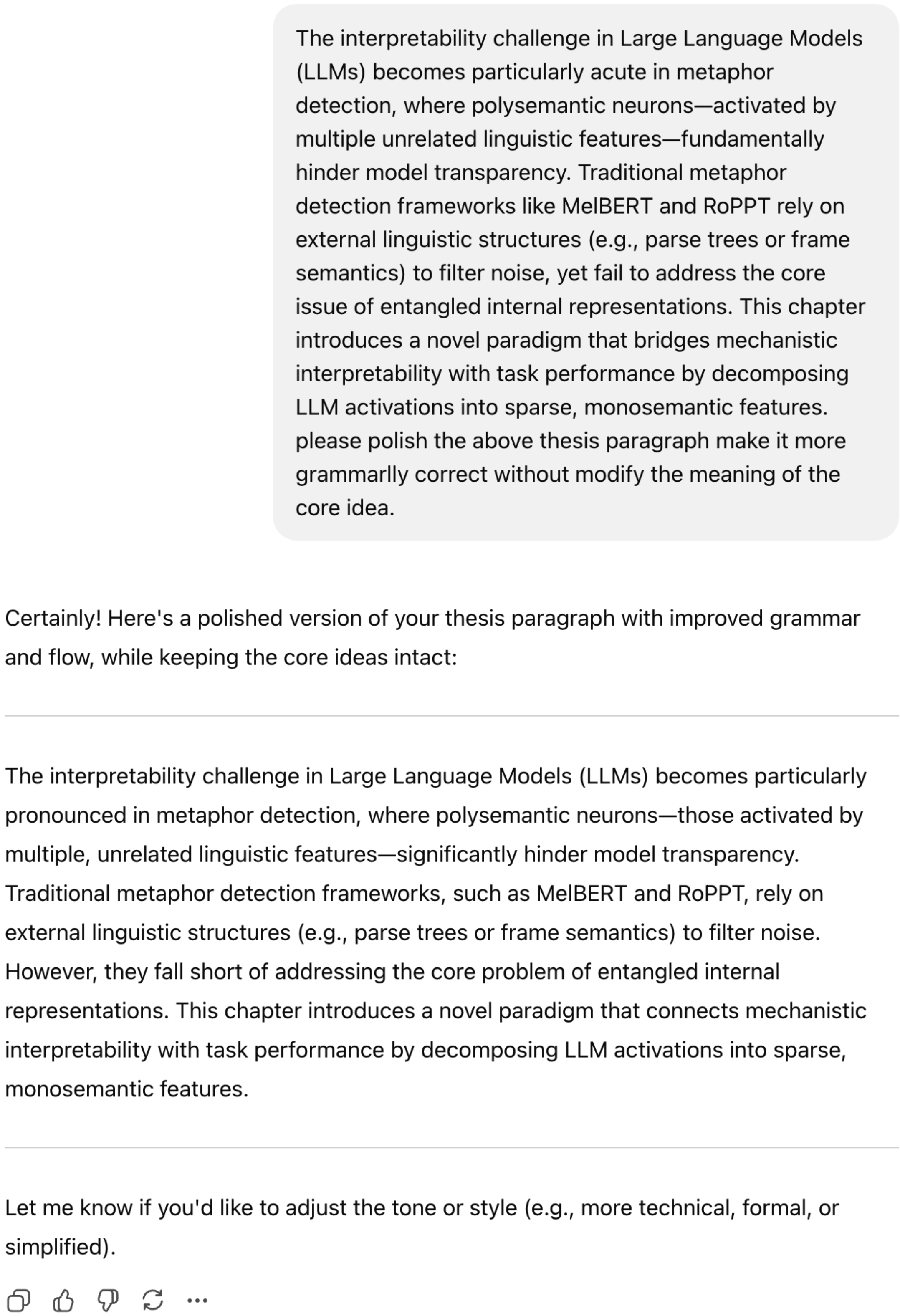
- Improving the clarity and fluency of written English in a small number of paragraphs.
- Paraphrasing or simplifying sentences for improved readability, while ensuring that all content and arguments remained my own.
- Using GenAI to identify and resolve LaTeX typesetting issues, including but not limited to table formatting, page layout control, and compatibility with language or font packages.

At no point was GenAI used to generate original content, conduct literature reviews, write arguments, or analyse data in thesis writing. The intellectual and analytical contributions in this thesis are entirely my own.

Evidence

Examples of interaction included:

All AI interactions took place in accordance with the University's data protection policies. No personal data or sensitive research material was input into the system.



The interpretability challenge in Large Language Models (LLMs) becomes particularly acute in metaphor detection, where polysemantic neurons—activated by multiple unrelated linguistic features—fundamentally hinder model transparency. Traditional metaphor detection frameworks like MelBERT and RoPPT rely on external linguistic structures (e.g., parse trees or frame semantics) to filter noise, yet fail to address the core issue of entangled internal representations. This chapter introduces a novel paradigm that bridges mechanistic interpretability with task performance by decomposing LLM activations into sparse, monosemantic features. please polish the above thesis paragraph make it more grammarlly correct without modify the meaning of the core idea.

Certainly! Here's a polished version of your thesis paragraph with improved grammar and flow, while keeping the core ideas intact:

The interpretability challenge in Large Language Models (LLMs) becomes particularly pronounced in metaphor detection, where polysemantic neurons—those activated by multiple, unrelated linguistic features—significantly hinder model transparency. Traditional metaphor detection frameworks, such as MelBERT and RoPPT, rely on external linguistic structures (e.g., parse trees or frame semantics) to filter noise. However, they fall short of addressing the core problem of entangled internal representations. This chapter introduces a novel paradigm that connects mechanistic interpretability with task performance by decomposing LLM activations into sparse, monosemantic features.

Let me know if you'd like to adjust the tone or style (e.g., more technical, formal, or simplified).

Figure 1: *Evidence of how to use GenAI to resolve LaTeX typesetting issues.*

\begin{figure*}[ht]
{figures/RoPPT/RoPPT_framework_r.png}
\caption{The overall framework of RoPPT. The parse tree of a sentence is reshaped to a target-oriented tree, and the context is pruned with a pre-set threshold. The sentence embedding is the average pooling result of hidden states for pruned context from RoBERTa. $\bigoplus$ denotes concatenation.}
\label{fig:roppt_framework}
\end{figure*}
the above latex figure exceeds the page width, please help me modify it to fit the page and center it directly.

6/6

To ensure your LaTeX figure fits the page width and is centered, you can switch from the `figure*` environment (which is for two-column layouts) to the standard `figure` environment, and also use `\centering` with `\includegraphics[width=\linewidth]` to scale it appropriately.

Here's your modified LaTeX code:

latex 复制代码

```latex
\begin{figure}[ht]
    \centering
    \includegraphics[width=\linewidth]{figures/RoPPT/RoPPT_framework_r.png
    \caption{The overall framework of RoPPT. The parse tree of a sentence
    \label{fig:roppt_framework}
\end{figure}
```

Notes:

- `\linewidth` ensures the image is scaled to fit within the current text column.
- `\centering` ensures the image is horizontally centered.
- `figure*` should only be used in two-column layouts when you want the figure to span both columns. If that is still your case, but the figure is too wide, consider reducing the width like `width=0.9\textwidth`.

Let me know if you're using a two-column layout and still want it to span both columns—I can help with further adjustments.

Figure 2: *Evidence of how to use GenAI to review the thesis for grammar and typographical errors.*