Cross-Lingual Multiword Expression Identification and Idiomaticity Prediction Using Autoregressive and Masked Language Models

by

Md. Arid Hasan

Bachelor of Science, Daffodil International University, Bangladesh, 2019

A Thesis Submitted in Partial Fulfilment of
the Requirements for the Degree of

Master of Computer Science

In the Graduate Academic Unit of Computer Science

Supervisor: Paul Cook, PhD, Computer Science

Examining Board: David Bremner, PhD, Computer Science, Chair
Roozbeh Razavi-Far, PhD, Computer Science
Shivam Saxena, PhD, Electrical and Computer Engineering

This thesis is accepted by the
Dean of Graduate Studies

THE UNIVERSITY OF NEW BRUNSWICK

May, 2025

Abstract

Token-level multiword expression (MWE) identification and idiomaticity prediction remain major challenges in natural language processing, demanding sophisticated approaches to address non-compositional meanings and idiosyncratic syntactic behaviors. These tasks involve identifying idiomatic expressions at the level of individual tokens, allowing systems to distinguish figurative from literal usages. This thesis explores cross-lingual MWE identification using the PARSEME 1.2 shared task dataset and idiomaticity prediction on the SemEval 2022 Task 2 dataset, where models are evaluated on unseen languages. We employ larger multilingual masked language models (MLMs), e.g., XLM-R and mT5, than previous work [137], which used supervised fine-tuning, and larger autoregressive models, e.g., GPT-4o, which previous work on these tasks have not considered. We adopted supervised fine-tuning of MLMs and autoregressive models and applied a prompt-based approach to autoregressive models. Our findings indicate that larger MLMs do not outperform the Swaminathan and Cook [137] results for the SemEval and PARSEME tasks, but that supervised fine-tuning of autoregressive models does.

Acknowledgements

Many people have supported me throughout this journey, and I would like to acknowledge a few of them, without whom this thesis would not have been possible. First of all, my family, the most important people in my life. I am deeply grateful to my parents for always listening and encouraging me, no matter the challenges. A special thanks goes to my wife, who always stayed by my side. Secondly, I would like to express my sincere gratitude to my supervisor, Dr. Paul Cook. His support and expertise have been invaluable throughout my thesis. Lastly, I would like to thank Dr. Firoj Alam and Mohammad Al-Munim for always encouraging me during difficult times and making this journey truly memorable.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Natural Language Processing (NLP) is a branch of artificial intelligence that allows machines to understand, generate, and communicate with human language using computational and linguistic techniques [69]. Common uses of NLP applications include question answering, machine translation, classification, and summarization [42].

Rule-based techniques that generally use hand-crafted grammars and syntax rules are widely adopted in earlier NLP research [124], such as machine translation [20]. However, probabilistic models and machine learning techniques have made progress by learning from large datasets [90]. The advancement in neural networks [47], especially word embedding techniques such as Word2Vec [92] and GloVe [101], improved text representation by better capturing semantic relationships. Moreover, the introduction of transformer architectures [144] that use self-attention mechanisms enabled systems to capture context more effectively across sequences. This advancement has enabled the large-scale development of language models, such as bidirectional encoder representations from transformers (BERT) [35] and generative pre-trained transformers (GPT) [109]. These models have significantly improved

performance in various NLP tasks, including text classification, machine translation, summarization, and question answering [42]. This development demonstrates that increased data availability, growing model complexity, and improved computational power have advanced NLP [47]. Despite these advancements, NLP still has to deal with challenges that include context-dependent meaning, ambiguity, and low-resource languages.[1] To address these issues, researchers have considered transfer learning and multilingual pretraining techniques [29].

Multiword expressions (MWEs) are word sequences that function as single units and possess unique linguistic properties that extend beyond compositional interpretations [123, 9]. MWEs cover a wide range of linguistic phenomena, such as idioms (e.g., *kick the bucket*), collocations (e.g., *coffee filter*), verb-particle constructions (e.g., *break out*), compound nouns (e.g., *bus stop*), and fixed phrases (e.g., *by and large*). The early approaches relied on hand-crafted lexicons and linguistic rules for the identification of MWEs [123], which were effective, however, these approaches required extensive manual labor to develop and maintain [32]. Later, statistical techniques were developed that used frequency-based metrics to identify MWEs, such as log-likelihood ratio and pointwise mutual information [38, 100]. The representation of MWEs in language resources has been improved by developing specialized lexicons and annotated corpora [145], such as treebanks [119]. Recent advances leverage deep learning models, such as BERT and ELMo, which incorporate contextual embeddings to improve the identification and prediction of MWEs [146].

MWEs pose challenges for several downstream NLP tasks because of their noncompositional meanings, irregular syntactic structures, and unpredictable usage patterns that cannot be directly determined from their component words [123]. MWEs may vary through inflection (e.g., *give up* $\rightarrow$ *given up*), changes in word order (e.g., *the*

[1]Low-resource language is defined by limited linguistic resources such as annotated data for various downstream NLP tasks.

meeting was called off → *they called the meeting off*), or idiomaticity (e.g., *kick the bucket* → 'to die'), which makes standard parsing and machine translation techniques ineffective [9]. For example, the phrase *kick the bucket* can be understood literally by combining the meanings of *kick*, *the*, and *bucket*, suggesting the action of 'kicking a bucket'. However, in an idiomatic context, *kick the bucket* has a different meaning, referring to 'to die'. Moreover, MWEs are difficult to interpret since they are context-dependent and their meanings might change depending on the surrounding text [112].

Token-level MWE identification involves labeling individual tokens or words within a text to determine whether they are part of an MWE [127, 45], which addresses the challenges of syntactic flexibility and diversity in these expressions. Based on their structure, MWEs can be categorized into various types [123], including light verb constructions (LVCs), where the verb is semantically bleached and often the expression means roughly the meaning as the verb (such as *take a nap*), verb-particle constructions (VPCs), which consist of a verb and a particle that modifies the meaning (such as *look up*), and noun compounds (NCs), where two or more nouns create a specific meaning (such as *coffee cup*). Many MWEs have idiomatic meanings that cannot be extracted from their individual words. Idiomaticity prediction leverages contextual information to distinguish whether a phrase is used literally or idiomatically [54]. MWEs can break standard compositional rules and show context-dependent syntactic or semantic behavior. Therefore, both token-level identification and idiomaticity prediction require advanced computational models.

Most MWE identification datasets are in English, while a few datasets are available in other languages such as Portuguese [34] and German [128]. To facilitate multilingual research, datasets such as PARSEME [127, 114, 115, 125] have recently been developed. Moreover, a multilingual dataset for idiomaticity prediction was recently developed in the SemEval 2022 Task 2 to support idiomaticity research [142]. We

used the datasets from SemEval 2022 Task 2 Subtask A and the PARSEME 1.2 edition to study idiomaticity prediction and MWE identification in this thesis. SemEval 2022 Task 2 Subtask A focuses on multilingual idiomaticity prediction that requires participants to distinguish instances of idiomatic expressions from literal ones across various languages using contextual information [142]. This promotes the development of cross-lingual and context-sensitive techniques, emphasizing the complexities of idiomatic expressions that can change meaning depending on cultural, linguistic, and situational factors. The PARSEME 1.2 shared tasks focus on token-level identification of verbal multiword expressions (VMWEs), extending earlier PARSEME tasks with enriched annotation guidelines and larger datasets for various languages [115]. Both endeavors are particularly significant because they address linguistic challenges such as semantic non-compositionality, syntactic variability, and the complexities of multilingual and cross-lingual analysis, which remain open problems in NLP.

MWE identification and idiomaticity prediction address the challenge of detecting MWEs and determining their meanings. In cross-lingual MWE identification and idiomaticity prediction, models are trained on one or more source languages to identify and interpret MWEs and idiomatic expressions in a target language, which is often low-resource [30, 41, 126]. This approach effectively generalizes linguistic information from resource-rich languages to low-resource languages by using multilingual knowledge transfer [141, 135] through sophisticated models such as multilingual BERT and XLM-R. Cross-lingual research is crucial for tasks like sentiment analysis and machine translation, where handling idioms or non-compositional words is important to preserve meaning across languages [9]. This approach could help to improve linguistic diversity, promote knowledge sharing, and significantly enhance the accessibility and equity of language technologies worldwide.

The study of Swaminathan and Cook [137] focused on cross-lingual MWE identifica-

tion and idiomaticity prediction using multilingual masked language models (MLMs). Their study showed that language models (such as multilingual BERT, XLM-R, and RoBERTa) can learn idiomaticity and MWE information from languages that are not seen during training. Furthermore, incorporating training data from other languages improves the overall performance of the models. [25] demonstrated that larger models could improve performance. Inspired by this, we aim to investigate whether larger MLMs (e.g., XLM-R-large) and autoregressive models (e.g., GPT-4o) can further improve performance on the SemEval and PARSEME tasks, particularly in cross-lingual settings. These models are trained on large multilingual corpora, which could be effective in MWE identification and idiomaticity prediction across the languages. In this study, we primarily explore two research questions (RQs) and two subquestions for the second question, as follows:

RQ1: *Do larger MLMs outperform the approach of Swaminathan and Cook [137] in cross-lingual settings?*

RQ2: *Do larger autoregressive models outperform the approach of Swaminathan and Cook in cross-lingual settings?*

RQ2.1: *Does the prompting-based approach of the large autoregressive models outperform the approach of Swaminathan and Cook in cross-lingual settings?*

RQ2.2: *Does supervised fine-tuning of larger autoregressive models outperform the approach of Swaminathan and Cook in cross-lingual settings?*

To perform this study, we used data from "Subtask A" of "SemEval 2022 Task 2: Multilingual Idiomaticity Detection and Sentence Embedding" and PARSEME edition 1.2 [115] on semi-supervised identification of verbal multiword expressions. We conduct experiments using XLM-R-large and mT5-base to address *RQ1*. The results in Chapter 5 indicate that larger masked language models for the SemEval

task could not consistently perform better than the approach of Swaminathan and Cook. Similarly, larger MLMs could not perform better on the PARSEME task. To address *RQ2, RQ2.1*, and *RQ2.2*, we choose GPT-4o and GPT-4o-mini for the autoregressive models. Our findings show that the prompting-based approach often performs better than the approach of Swaminathan and Cook on the SemEval task using GPT-4o, but not using GPT-4o-mini. However, supervised fine-tuning of both autoregressive models performs better than the approach of Swaminathan and Cook in most cases on this task. The prompting-based approach for the PARSEME task could not perform better than the approach of Swaminathan and Cook. However, supervised fine-tuning of large autoregressive models does perform better than Swaminathan and Cook in this task.

The main contributions of this thesis are as follows:

- We evaluated fine-tuned larger MLMs such as XLM-R-large and multilingual T5 compared to the study of Swaminathan and Cook [137].
- We developed prompts for SemEval and PARSEME tasks and evaluated prompting-based approach for autoregressive models without training.
- We prepared instruction-following datasets to apply and evaluate supervised finetuning of GPT-4o and GPT-4o-mini.

The thesis is structured as follows. Chapter 2 provides a detailed background on language models, focusing on masked language models and autoregressive models, and multiword expressions. Chapter 3 describes the models that we used in this study. We discuss the datasets, experimental setup, implementation details, parameter settings, supervised fine-tuning and prompting-based approach for autoregressive models, and evaluation metrics in Chapter 4. Chapter 5 presents and discusses the results and compares them with the Swaminathan and Cook [137] results. Finally,

we conclude by summarizing key findings and providing potential future research directions in Chapter 6.

Chapter 2

Related Work

This chapter discusses the prior research on language models (LMs). We explore the transformer architecture and transformer-based masked LMs, such as XLM-RoBERTa. We further explore transformer-based autoregressive models such as generative pretrained transformer (GPT) models, e.g., GPT-4o and GPT-4o-mini. We then continue exploring multi-word expressions (MWEs), their challenges for natural language processing (NLP), and the types of NLP research that have been done in the past for MWEs. Finally, we conclude this chapter by discussing the previous research on the prediction of cross-lingual idiomaticity and emphasizing the cross-lingual token-level identification of MWEs.

2.1 Language Models

Language models have evolved from statistical techniques to large scale generative models using deep learning-based architectures such as recurrent neural networks and transformers. This section reviews key developments in language modeling, focusing on autoregressive models, transformers, GPT, masked LMs, and fine-tuning these

models.

2.1.1 Historical Development of Language Models

Language models (LMs) are one of the fundamental components of NLP that enable machines to understand, generate, and summarize natural languages. These models have seen substantial development, advancing from simple n-gram models to state-of-the-art deep learning architectures such as transformers. Early studies on language models were based on statistical methods that included n-grams, hidden Markov models, and rule-based methods. N-gram models, which often encounter problems with data sparsity and an inability to capture long-range dependencies, use conditional probability to calculate the likelihood of word sequences [16]. Long-range dependencies refer to relationships between elements in a sequence that are far apart, and in tasks like language processing and time-series analysis, models need to capture these dependencies to understand context effectively. The hidden Markov model introduced latent variables for better sequence modeling but encountered issues in capturing long-range dependencies [108]. Moreover, rule-based language models rely on a set of predefined rules to make predictions. These models have been studied to handle complex event processing and reasoning with the use of an expressive logic-based technique [4]. However, rule-based methods struggle to adapt to changing data and generalize to new scenarios [161].

In NLP, a "word" is typically a sequence of characters separated by spaces or punctuation; however, defining a word can be complex due to variations in languages, contractions (e.g., *don't*), and compound words (e.g., *New York*). Traditional word representation techniques, such as one-hot encoding and TF-IDF, are unable to capture semantic relationships. The introduction of word embeddings, including Word2Vec [91], GloVe [101], and fastText [14], revolutionized NLP by improving the encoding

of words in dense vector spaces that can capture semantic relationships. The use of neural networks in NLP has significantly improved language modeling. Bengio et al. [11] introduced a feedforward neural network to learn word embeddings and predict the next word in a sequence, which outperforms traditional n-gram models. However, the feedforward neural network often forgot the previous input because of a fixed context size. The recurrent neural network (RNN), which can retain information from previous inputs, was introduced in [122] to overcome this issue. However, one issue with RNN-based models is the vanishing gradient problem [58], which occurs when the gradient becomes insufficient to update the network's hidden state as the input size increases. The gradient is used in optimization problems to guide the process of minimizing a loss or cost function. To overcome this issue, long short-term memory [59] and gated recurrent units [26] were introduced, which incorporated memory methods that allow only important information to be retained for sequential data processing.

The study of Cho et al. [22] found that longer input sequences reduce RNN performance compared to RNN models trained on shorter sequences. To overcome the performance issue of RNN and understand context in longer sentences, Badhanau et al. [7] introduced the attention mechanism. The attention mechanism allows the model to focus on the relevant parts of the input data. Later, Luong et al. [88] proposed an improved attention mechanism that incorporates both global attention, which attends to (refers to a token looking at or drawing information from other tokens) all source words by default, and local attention, which attends to a subset of source words.

The breakthrough in language modeling came with the introduction of the Transformer architecture by Vaswani et al. [144]. Transformers use self-attention methods (discussed in Section 2.1.3) to effectively capture contextual dependencies and adapt internal parameters to focus on the most important words for each position. This

breakthrough resulted in the development of large-scale language models such as Bidirectional Encoder Representations from Transformers (BERT) [35], generative pretrained transformer (GPT) [109, 110, 17], and Text-to-Text Transfer Transformer [111]. Pretraining these models involves training on large text corpora using self-supervised learning (discussed in Section 2.1.4), while fine-tuning adapts the model for specific downstream tasks. BERT introduced bidirectional contextual learning, improving performance on downstream NLP tasks, while GPT models employed autoregressive generation to achieve state-of-the-art results in text generation.

2.1.2 Autoregressive Language Models

Autoregressive language models have become one of the fundamental foundations of NLP and modern artificial intelligence. These models are more capable of generating text than masked language models by predicting one token at a time based on the previous tokens,[1] making them very effective for various downstream NLP tasks, such as text generation, translation, and summarization [17]. Autoregressive LMs have developed substantially from simple statistical methods such as n-gram LMs to complex deep learning architectures such as GPT. Autoregressive language models estimate the likelihood of a word sequence X by factoring the joint probability distribution over tokens:

$$P(X) = \prod_{t=1}^{n} P(x_t|x_1, x_2, ..., x_{t-1}) = \prod_{t=1}^{n} P(x_t|x_{<t}) \tag{2.1}$$

where the likelihood of each token x_t depends on all tokens that came before it [17]. Although this sequential method allows for the generation of fluent text, it increases

[1]Tokens are the individual units that result from tokenization, the process of breaking text into smaller components; they can be words, subwords, or even characters, depending on the tokenization approach. For example, in subword tokenization (used in models like BERT [35]), a single word might be split into multiple tokens (e.g., "unhealthy" → ["un", "healthy"]).

inference latency and limits parallelization.

Early autoregressive models, such as n-gram models [16] and hidden Markov models [108], suffered from sparsity and limited contextual understanding.[2] These models employed the Markov assumption, where a word's likelihood was determined solely by a limited history of prior words, resulting in a limited ability to capture long-range dependencies. Neural network-based autoregressive language models, such as RNNs, improved sequence modeling by capturing long-range dependencies [11]. Although the recurrent structure of RNNs enabled sequential data processing, these models struggled with vanishing gradient problems [58]. Long short-term memory networks [59] and gated recurrent units [22] developed gating methods to limit information flow, mitigate vanishing gradients, and improve context understanding. However, these models have scalability issues because of their sequential nature, which makes them inefficient for training on very large amounts of text.

The transformer model [144] led to major advances in autoregressive language models by incorporating self-attention mechanisms, which enabled efficient parallel processing while preserving contextual awareness. The addition of position-aware feedforward layers and multihead attention substantially improved the model's ability to learn dependencies across long-range sequences (details on transformer architecture, self-attention mechanisms, and multihead attention are discussed in Section 2.1.3). This led to the development of state-of-the-art autoregressive language models such as GPT models [109, 110, 17]. GPT models use only autoregressive decoding to generate text. GPT-3 [17] showed remarkable zero- and few-shot learning capabilities, particularly influencing downstream NLP tasks with the use of a causal self-attention mechanism, where each token attends only to the previous input tokens, ensuring an autoregressive, i.e., left-to-right, structure. Recent developments include GPT-

[2]Sparsity refers to the presence of a lot of zero values in a representation, meaning most elements contain no meaningful information.

4, which incorporates multimodal capabilities to enhance contextual knowledge and reasoning [2].

2.1.3 Transformers

Transformers is a deep learning based architecture proposed by Vaswani et al. [144], which fundamentally influenced the rapid advancement of NLP. Previously, recurrent neural networks (RNNs) and long short-term memory (LSTM) were widely used for sequential modeling in neural networks. Transformers were proposed as an alternative because of limitations of RNNs and LSTM, such as inefficiency in capturing long-range dependencies and computational inefficiency in parallelization [7, 59]. Unlike earlier neural networks such as RNNs and LSTMs, transformers use attention mechanisms extensively, which enables better processing of sequential data [144]. In the transformer model, the input is encoded using multiple encoder blocks, while the output is generated using multiple decoder blocks. The encoder efficiently captures contextual information for each word while creating representations of the input. This is accomplished by creating a fixed length context vector that summarizes the encoded data. Then, the decoder leverages the encoder's hidden state to produce the output sequence. The attention mechanism is a key component of the transformer architecture, which improves the model's capacity to attend to the relevant parts of the input when making predictions.

Figure 2.1 shows an encoder and decoder block of the transformer architecture. Both the encoder and decoder are built using multiple identical layers, each containing two primary sub-layers: a multi-head self-attention mechanism and a feedforward network. Each sub-layer is followed by residual connections and layer normalization. Although the conventional attention mechanism computes the hidden states using the complete input sequence, self-attention also addresses various related positions

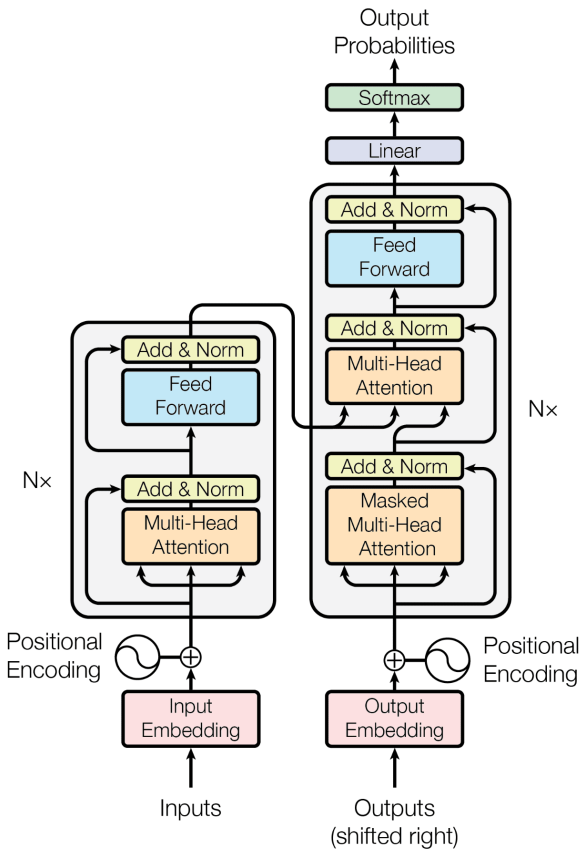

Figure 2.1: Encoder and decoder representation of transformer architecture taken from [144].

within the sequence, enabling the capture of word dependencies. For instance, in the sentence *"the cat sat on the mat because it was comfortable"* when processing the word "it", the self-attention mechanism identifies that "it" refers to "the mat" rather than "the cat", by computing attention scores between words and capturing dependencies across different positions.[3] Multi-head attention refers to the blocks of attention layers found in transformers. Instead of computing a single attention score, multi-head attention runs multiple attention operations in parallel, each with different learned projections. This enables the model to capture various relationships across positions in a sequence. The model can better capture the aforementioned

[3]Attention scores are computed by calculating the similarity between a query vector and a set of key vectors using a dot product.

dependencies in the input sequence with the aid of these attention blocks. The attention blocks analyze the entire input sequence to identify the most relevant words for each word in the sequence. The encoder processes inputs using self-attention layers, while the decoder generates output using the same technique.

Transformers performed better than neural network models in tasks such as machine translation [144]. Moreover, it improves performance on various downstream tasks such as text classification [155, 160], image recognition [99, 56, 36], question answering [95, 147], text summarization [53, 75], sequence labeling [18], and information retrieval [43]. Since we used GPT and BERT models in this study which are both based on transformers, we give a detailed description of these models.

2.1.4 GPT Models

GPT models have demonstrated substantial breakthroughs in artificial intelligence (AI), particularly in NLP. These models, based on the transformer architecture introduced by Vaswani et al. [144], have demonstrated remarkable capabilities in text generation [159], question answering [116], and conversational AI [21]. GPT models use self-supervised techniques in the pretraining stage, employing large-scale text datasets to generate human-like text.

In 2018, OpenAI introduced GPT-1 as the first generative pretrained transformer model to demonstrate the effectiveness of pretraining a transformer architecture on large-scale text data and then fine-tuning it for specific tasks [109]. Unlike BERT [35], which was bidirectional, GPT-1 was designed as a unidirectional (left-to-right) transformer decoder and trained on the BookCorpus dataset [165] and fine-tuned on specific NLP tasks, demonstrating the effectiveness of transfer learning in NLP. This approach significantly reduced the need for large labeled datasets. Figure 2.2 shows the GPT architecture, which uses a masked multi self-attention layer and a

feedforward network layer. Masked multi self-attention enables models to capture contextual relationships by allowing words to attend selectively to other words, with masking restricting future word access in GPT. Each layer is followed by residual connections and layer normalization. GPT-2 was developed based on its predecessor by increasing the model size to 1.5 billion parameters [110]. GPT-2 demonstrated zero-shot capabilities, performing tasks on which it was not explicitly trained. GPT-3 was introduced in 2020, which expanded the scale and complexity of the model (for example, expanding to 175 billion parameters from 1.5 billion and training on a diverse corpus of 570GB of text data using 96 attention layers) to achieve remarkable improvements in text generation and comprehension [17]. GPT-3 employs a self-supervised pretraining strategy to predict next-word tokens in vast amounts of unlabeled text, demonstrating remarkable ability on zero-shot, one-shot, and few-shot learning scenarios compared to earlier models. In self-supervised learning, the model learns from unlabeled data by creating its own supervision signals. GPT-3

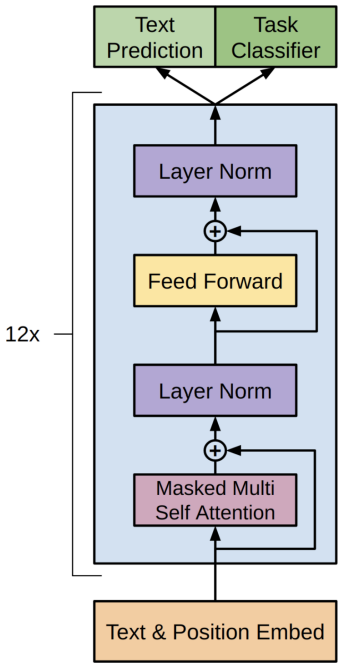

Figure 2.2: GPT architecture taken from [109] which only uses the transformer decoder and training objectives.

can perform tasks with instructions or with a few examples in its contexts without supervised fine-tuning (discussed in Section 2.1.5 below).

GPT-4 is a recent development that demonstrates significant improvements in performance, reasoning capabilities, and accuracy (claim of OpenAI [2]). This model incorporates multimodal abilities and is capable of accepting both textual and visual inputs. Moreover, GPT-4 improved the ability to address limitations of GPT-3, including context-sensitive errors, hallucinations, and limited multimodal understanding [2]. Although specific architectural details and model size have not been fully disclosed, GPT-4 uses the transformer decoder-based model with improved optimizations, extended context length, and substantial fine-tuning through reinforcement learning with human feedback.

2.1.5 Fine-tuning GPT Models

Pretrained language models such as GPT and BERT are initially trained on vast text corpora in a self-supervised manner. However, fine-tuning is needed for specific applications such as question answering, text summarization, and sentiment analysis, where models are trained on task-specific datasets using supervised learning. Fine-tuning improves model accuracy, reduces bias, and improves response quality in specific contexts [61]. Supervised fine-tuning is the process of adapting a pretrained model to a specific task by training it on labeled data. The process starts with a general model that has been trained on a large dataset, such as BERT for language understanding or GPT for text generation. The model is then given a task-specific dataset in which each input is paired with the correct output, allowing it to learn through supervised learning. During fine-tuning, the model's weights are adapted using optimization techniques such as backpropagation to minimize errors and improve accuracy. Radford et al. [109] demonstrated the potential of supervised

fine-tuning in combination with self-supervised pretraining, developing a system that uses vast amounts of data to learn language representations. The size of the models has increased with the development of GPT-2 [110], GPT-3 [17], and GPT-4 [2], improving their capacity for language generation and understanding [93].

Early methods investigated comprehensive model fine-tuning, in which supervisory signals are used to update the entire network [157]. Although this approach is efficient for small models, it is computationally costly, especially when dealing with models that have billions of parameters, and it raises the possibility of overfitting when the target dataset is small [111]. To address these difficulties, researchers have developed parameter-efficient fine-tuning techniques. Notably, adapter-based fine-tuning [60] involves preserving most of the model's pretrained parameters while adding small bottleneck layers inside each transformer block. Li and Lian et al. [82] presented prompt tuning, another well-known technique where learnable prompt embeddings are used to condition the model's behavior, allowing task adaptation without substantial changes to the model's underlying architecture. Similarly, Hu et al. [62] presented low-rank adaptation, which efficiently fine-tunes models with few parameter changes by breaking down weight updates into low-rank matrices.

Optimization strategies that enable stable model convergence during fine-tuning have recently been extensively studied. Regularization techniques like weight decay [85] and dropout [133] have been important in reducing overfitting and improving the model's generalization capabilities. Moreover, the use of sophisticated optimizers such as AdamW improved fine-tuning stability even when models are adapted to domains that differ greatly from pretraining data [50]. Hinton et al. [57] use knowledge distillation, in which a smaller student model is trained from a fine-tuned teacher model to reduce computational cost. Researchers use augmentation approaches such as back-translation, which uses machine translation to generate paraphrased training examples, to address data scarcity in supervised fine-tuning [129]. GPT models are

used to generate synthetic data, which adds artificial training samples to the original data [3]. Supervised fine-tuning has enabled significant advancement in multiple areas, such as classification [138, 134], generation [81], and conversation [163]. Supervised fine-tuning is an important step in adapting GPT-like models to real-world scenarios. Developing parameter-efficient tuning methods, including self-supervised learning, and reducing bias (such as balancing among labels) in labeled datasets are the three main goals of future research to improve the efficiency of fine-tuning [52].

2.1.6 Masked LMs

Pretrained language models have advanced significantly in the last few years in the field of NLP. Early work in this field, such as ELMo [102] and generative models like GPT [109], paved the way by capturing contextual dependencies in text. However, the introduction of Masked Language Models (MLMs) such as BERT [35] resulted in a paradigm change. Unlike autoregressive models that generate text sequentially, MLMs use bidirectional context by randomly masking portions of the input and predicting the masked tokens. This technique promotes a more in-depth knowledge of language structure and semantic nuances [153] and has become the foundation for many subsequent innovations in language modeling. Moreover, this technique, first popularized by BERT [35], has proven useful in various downstream NLP tasks, including text classification [44, 134], named entity recognition [136], and question answering [154].

Early versions of BERT used WordPiece tokenization [132], which divides words into subword units to mitigate out-of-vocabulary problems. Later models investigated byte-pair encoding [130] and SentencePiece [74], which is language independent subword tokenizer, improves language coverage and fine-grained representation of subword units. The choice of tokenization directly impacts the model's ability

to learn semantic relationships, particularly in morphologically rich or low-resource languages.

BERT's training process consists of two major stages: pretraining and fine-tuning. The pretraining phase is self-supervised, indicating that it does not require labeled data. However, it is based on two main goals: masked language modeling and next sentence prediction. MLMs primarily rely on the transformer architecture [144], which leverages self-attention mechanisms to model complex dependencies across all tokens in a sequence. In MLM training, a fixed percentage (typically 15%) of tokens is selected for masking. During pretraining, a token is either replaced by a special *[MASK]* token, substituted with a random token, or left unchanged, a strategy that introduces noise and encourages robust contextual learning [35]. This strategy is essential to prevent overfitting to fixed masked positions and enable the model to learn a more general language representation.

The second goal, next sentence prediction, entails training BERT to determine whether a given sentence pair is sequentially related or randomly selected. This helps improve performance on tasks that require sentence-level understanding. Following the completion of pretraining, labeled datasets are used to fine-tune the model for specific downstream tasks. Fine-tuning involves adding a task-specific classification layer on top of the pretrained BERT model and optimizing it using task-specific objectives, such as cross-entropy loss for classification tasks. The key benefit of this process is transfer learning, which allows BERT's contextualized knowledge from large-scale pretraining to be applied to a wide range of NLP applications with relatively little labeled data.

Since BERT was introduced, several variants and improvements have been proposed for MLMs. Significant performance improvements were obtained on a number of downstream tasks when RoBERTa [84] improved the pretraining process by eliminat-

ing the next sentence prediction task and adding dynamic masking, where different tokens are masked each time a sentence is processed by the model. ALBERT [77] introduced cross-layer parameter sharing and factorized embedding parameterization to reduce the overall parameter count while maintaining performance. ELECTRA [28] used a different approach, which trained a discriminator to recognize the substituted tokens after substituting them with plausible alternatives developed by a small generator. This "substituted token detection" objective yields higher sample efficiency because every token is either classified as real or fake, unlike the conventional MLM setting, where only a fraction of tokens is predicted. Additionally, models such as SpanBERT [68] mask contiguous spans rather than individual tokens to better capture phrasal-level representations.

MLMs have also been extended to multilingual and domain-specific contexts. Multilingual BERT is trained on a diverse corpus of texts from multiple languages, enabling cross-lingual transfer and resource sharing. This idea is further scaled by XLM-R [30], an evolution of mBERT, which uses a larger multilingual corpus derived from several language families for training. In parallel, BERT has been pretrained on specific domains such as scientific and biomedical literature, respectively, to create domain-specific models such as SciBERT [10] and BioBERT [79]. These modifications show that MLMs can be adapted for specialized applications where domain-specific subtleties are crucial, in addition to the ability to understand language that describes a category or group as a whole, rather than specific individuals (such as *all dogs bark* and *birds have wings*).

2.1.7 Fine-tuning Multilingual BERT Models

Fine-tuning pretrained MLMs, particularly transformer-based architectures such as BERT [35], has become standard practice in NLP due to its efficacy in achieving

state-of-the-art performance across various downstream tasks. The introduction of multilingual transformer models, notably XLM-R [29] and multilingual text-to-text transfer transformer (mT5) [151], has significantly advanced the field by enabling effective cross-lingual transfer and multilingual understanding.[4] These models use the transformer framework to learn complex language-independent representations from large multilingual datasets, building on the success of previous MLMs. By addressing limitations in language coverage and fine-tuning capability for various NLP tasks, both XLM-R and mT5 have driven progress on benchmarks such as cross-lingual natural language inference [30], translation [89, 67], summarization [1], and question answering [12].

XLM-R is a multilingual variant of RoBERTa [84] that uses an encoder-only transformer architecture and is trained on a large dataset generated from CommonCrawl, covering more than 100 languages. Its training approach enables the model to learn complex representations even for low-resource languages by using a MLM objective without depending on next sentence prediction. Conneau et al. [29] demonstrated that XLM-R performs better than previous multilingual models such as mBERT on various cross-lingual benchmarks, including XNLI [30], MLQA [80], and TyDiQA [27]. Fine-tuning XLM-R for downstream NLP tasks involves adding task-specific layers on top of the pretrained transformer encoder and updating all parameters in the whole model using labeled task-specific data [63]. This approach performs better than previous multilingual models in tasks such as named entity recognition, sentiment analysis, and question answering [106, 150]. The effectiveness of fine-tuning XLM-R in low-resource and zero-shot cross-lingual transfer scenarios has been explored in several studies [5, 23, 96]. For instance, the zero-shot transfer capabilities of XLM-R explored by Artetxe et al. [5] showed that fine-tuning on high-resource languages significantly improves performance on low-resource lan-

[4]Multilingual understanding refers to the ability of a system to understand and generate meaningful responses across various downstream NLP tasks in multiple languages.

guages without explicit cross-lingual supervision. The cross-lingual transferability of XLM-R embeddings was further investigated by Lauscher et al. [78], emphasizing the model's robustness and generalization capabilities across typologically diverse languages. Furthermore, Pfeiffer et al. [103] proposed AdapterFusion, a method for integrating language-specific adapters into XLM-R, allowing efficient fine-tuning and improved cross-lingual transfer performance.

The T5 design [111] is extended to multilingual environments by mT5, which was introduced by Xue et al. [151]. Unlike encoder-only models like XLM-R, mT5 employs an encoder-decoder architecture, which makes it suitable for generative tasks such as translation, summarization, and question answering. Training on the mC4 dataset [151], which is a multilingual version of the C4 corpus [111], gives mT5 a broader linguistic exposure that improves its generalization across many language families. Fine-tuning mT5 involves developing downstream NLP tasks as text-to-text problems, where the model generates the desired output based on the input text. Xue et al. [151] showed that mT5 achieves better performance on various multilingual benchmarks, including XTREME [63] and XQuAD [5]. In recent studies, fine-tuning mT5 for multilingual summarization [53] and translation [24] tasks has been explored, achieving state-of-the-art results on both tasks. mT5 has been studied for multilingual question answering [70], emphasizing its effectiveness in generating accurate and contextually relevant answers across multiple languages.

Several studies have also focused on improving the efficiency and effectiveness of fine-tuning multilingual models. Houlsby et al. [60] introduced adapter modules, which are lightweight neural components inserted into transformer layers, enabling efficient fine-tuning of large multilingual models such as XLM-R and mT5. A modular adapter-based framework [104] was introduced to extend this approach, particularly reducing computational costs while maintaining high performance in multilingual transfer tasks. Moreover, parameter-efficient fine-tuning methods such as

prefix-tuning and prompt-tuning were explored, demonstrating their effectiveness in adapting multilingual models to downstream tasks with minimal computational overhead [121].

2.2 Multiword Expressions

We discuss the work done on multiword expressions (MWEs) in the context of NLP in this section. MWEs are an essential aspect of NLP and linguistics, encompassing phrases whose meanings cannot be derived from their individual components [9]. These expressions include idioms (e.g., *bite the bullet*), collocations (e.g., *coffee filter*), phrasal verbs (e.g., *break out*), and compound nouns (e.g., *coffee cup*). Identifying and processing MWEs accurately is critical for various NLP applications [31], including machine translation, information retrieval, and sentiment analysis. Token-level MWE identification and idiomaticity prediction are the two NLP tasks related to MWEs that we primarily address in this section. Before delving into these topics, we first provide a brief definition of MWEs along with examples. Then we discuss the importance of MWEs in NLP and the difficulties that language models encounter with respect to them.

2.2.1 Definition and Examples

MWEs require specialized methods for identification and processing in NLP. MWEs are lexical items consisting of more than one word, but which function as a single unit in terms of meaning, syntax, or usage [9]. These expressions defy simple word-by-word interpretations and frequently contain a high degree of idiomaticity [149]. Many MWEs are idiomatic, which means they convey figurative meanings that cannot be completely predicted from their constituent words [65]. For instance, *kick the bucket*

means ‘to die’ which is obviously unrelated to the literal meaning of actually ‘kicking a bucket’. We explore idiomaticity prediction in the SemEval 2022 shared task 2 subtask A.

Early studies on MWEs focused on their linguistic characteristics and classifications. There are several different types of MWEs. Sag et al. [123] categorized MWEs based on their syntactic and semantic properties, emphasizing the challenges in their computational processing. Noun compounds (NCs) are the most common type of MWE, in which two or more nouns come together to create a MWE (such as *coffee cup*). Baldwin et al. [8] initially proposed utilizing latent semantic analysis to estimate the compositionality of MWEs, particularly focusing on noun compounds. Later, [71] expanded on this by suggesting an unsupervised method to predict the compositionality of noun compounds. [117] proposed a dataset to analyze noun compounds and determine how each component contributes to the meaning. The majority of noun compound research worked on type-level predictions, where compositionality is assessed as a general property of an MWE. For instance, when we predict the idiomaticity of the expression itself, this is known as type-level idiomaticity prediction. However, it is also crucial that we make token-level predictions, where compositionality is determined based on a specific usage of the expression in context. [41] demonstrated that the model’s capacity to predict the compositionality of noun compounds is enhanced when noun compounds are predicted at the token level.

This study uses the multilingual dataset from PARSEME 1.2 shared task [115], containing sentences with token-level annotation for verbal MWEs (VMWEs). VMWEs are MWEs where the main word of the expression is a verb. In this dataset, the tokens are annotated into nine categories of MWEs, which are inherently reflexive verbs, full light verb constructions, causative light verb constructions, verbal idioms, fully non-compositional verb-particle constructions, semi non-compositional verb-particle constructions, multi-verb constructions, inherently adpositional verbs,

and inherently clitic verbs.[5] The light verb constructions (LVCs) are formed by a verb and a noun, which either directly depend on the verb or are introduced by a preposition. The definitions are taken from Ramisch et al. [114] while we changed the examples into English, where these examples are also taken from the detailed annotation guidelines presented by Ramisch et al. [114] as follows:

- **Inherently Reflexive Verbs (IRV):** pervasive in Romance and Slavic languages, and present in Hungarian and German, in which the reflexive clitic either always co-occurs with a given verb, or markedly changes its meaning or sub-categorization frame, such as *to **find oneself** in a difficult situation.*
- **Full Light Verb Constructions (LVC.full):** LVCs in which the verb is semantically totally bleached, such as *The party **gave priority** to senior members.*
- **Causative Light Verb Constructions (LVC.cause):** LVCs in which the verb adds a causative meaning to the noun, such as *the new law **provoked** the **destruction** of the building.*
- **Verbal Idioms (VID):** Groups all VMWEs that do not belong to other categories, and most often have a relatively high degree of semantic non-compositionality, such as 'kick the bucket' → 'to die'.
- **Fully non-compositional Verb-Particle Constructions (VPC.full):** In which the particle totally changes the meaning of the verb, such as 'to do in' → 'to kill'.
- **Semi non-compositional Verb-Particle Constructions (VPC.semi):** In which the particle adds a partly predictable but non-spatial meaning to the verb, such as 'to eat up' → 'to eat completely'.

[5] A clitic is a short word (often a pronoun or auxiliary verb) that behaves like a word grammatically but is phonologically dependent on another word.

- **Multi-Verb Constructions (MVC):** Close to serial verbs which are semantically non-compositional in Asian languages like Chinese, Hindi, Indonesian, and Japanese (but also attested in Spanish), such as ***Hindi: baiTh ja*** (*English:* 'sit go') → 'sit down'.

- **Inherently Adpositional Verbs (IAV):** Include idiomatic combinations of verbs with prepositions or post-positions, depending on the language, such as 'to **count on**' → 'to depend on'.

- **Inherently Clitic Verbs (LS.ICV):**[6] In which at least one non-reflexive clitic either always accompanies a given verb or markedly changes its meaning or its sub-categorization frame, such as ***Italian: prenderle*** (*English:* 'take-them') → 'get beaten up'.

2.2.2 Overview of NLP Research on MWEs

The complex nature of MWEs and their significance for various NLP applications, such as sentiment analysis, information extraction, and machine translation, have long been acknowledged by researchers [123]. During the past two decades, various types of MWE research have emerged, focusing on topics such as extraction, classification, lexical representation, and integration into downstream NLP tasks. Early research focuses on the extraction and identification of MWEs from large text corpora, employing symbolic approaches, such as handcrafted lexicons and linguistic rules, to detect idiomatic expressions or collocations [123]. Although these methods were valuable for capturing highly idiomatic or domain-specific MWEs, they often required extensive manual labor and linguistic expertise [32]. Subsequent work employed statistical methods that leveraged frequency-based association measures, such

[6]Language-specific (LS) categories are carefully defined and accompanied by linguistic tests that allow to distinguish them from other categories.

as pointwise mutual information or log-likelihood ratio, to identify candidate MWEs [38, 100].

Identifying and classifying MWEs based on their syntactic and semantic characteristics is another important research area. MWEs can vary from semi-fixed phrases that allow some inflection or substitution (such as *spill the beans*, which can become *spilled the beans*) to fixed expressions with nearly no syntactic flexibility (such as *by and large*) [123]. The semantic compositionality of MWEs varies widely, from relatively transparent meanings (e.g. *coffee cup*) to idiomatic expressions (such as *kick the bucket*) whose meanings are not predictable from individual words [112]. Researchers have proposed several taxonomies and metadata frameworks to better capture the nuances of MWEs across languages because of this variability [140]. Another key area of focus is lexical representation and resource construction for MWEs. Standard lexical databases often fail to accurately represent MWEs, since they do not always follow a regular lexical composition. Therefore, specialized lexicons and annotated corpora have been developed to represent the linguistic idiosyncrasies of MWEs [86, 145]. These resources include treebanks with comprehensive annotations of MWEs in context and lexicons including idioms or light verb constructions [119, 145].[7]

MWEs are difficult to recognize because of their potential idiomatic behavior [126]. However, because of the large and increasing number of MWEs, identifying and extracting them is important [40, 114, 115, 142]. Moreover, it is important to identify MWEs to improve model performance for various downstream NLP tasks, such as machine translation. In machine translation, for example, specialized MWE handling mechanisms can significantly improve the accuracy and fluency of translated output, as literal translations of idiomatic expressions sometimes provide non-sensical or mis-

[7] Treebanks are linguistic datasets that contain sentences annotated with syntactic or grammatical structure, usually in the form of parse trees.

leading results [15]. Moreover, bilingual MWEs are defined by Ren et al. [118] as MWEs that translate words from a source language into a target language exactly one-to-one, indicating literal translation. A biLSTM-based model was proposed to translate MWEs from a source language to MWEs in a target language in [158]. Similarly, in information extraction and sentiment analysis, identifying multiword terms (for example, domain-specific compound nouns) can improve the accuracy of entity recognition and sentiment classification [94]. Moreover, specialized MWE modules have been shown to reduce parsing errors in syntactic parsing by considering certain sequences as single lexical units [33]. The growing integration of modules that are aware of MWEs into end-to-end neural architectures underscores their importance to achieve state-of-the-art performance in various NLP tasks [19]. More recently, researchers have used machine learning algorithms and neural network models, using large corpora and contextual word embeddings (such as BERT [35] or ELMo [102]) to detect and classify MWEs more robustly [146]. Researchers are investigating these advanced computational techniques to improve results in various downstream tasks, such as machine translation, information retrieval, and sentiment analysis [32].

2.2.3 Cross-lingual Idiomaticity Prediction (SemEval)

The cross-lingual prediction of idiomaticity has developed as a challenging and important research area in NLP, especially as the amount of multilingual written content increases and systems are required to interpret, translate, or otherwise process idiomatic expressions in diverse languages [9]. Idiomatic expressions are non-compositional in meaning, which means that their overall meaning cannot be determined immediately from the literal interpretation of their constituent words [123]. These characteristics make NLP tasks such as information retrieval, machine translation, text categorization, and question answering extremely challenging [40]. There-

fore, there is an increasing amount of research on how to effectively identify and interpret MWEs, such as idioms and phrasal verbs, in a multilingual setting [32, 51]. Early approaches in idiomaticity prediction often relied on large phrase repositories, part-of-speech tag patterns, and statistical measures of cooccurrence or semantic cohesion [32]. Although such techniques were moderately successful in monolingual contexts, they often struggled to extend to multiple languages or more flexible idiomatic constructions [9].

Initial efforts to represent idiomatic phrases using non-contextual embeddings focused on extracting frequently occurring n-grams (e.g., *big fish*) from text and then learning their representations based on surrounding context [92]. However, the efficiency of this strategy decreases dramatically as the length of the idiomatic phrase increases because of data scarcity [34]. Recent research demonstrates that even state-of-the-art pretrained contextual models, such as BERT, are unable to effectively represent idiomatic expressions [41]. To address these challenges, Madabushi et al. [142] organized a shared task at SemEval 2022, aiming to detect and represent MWEs that are possibly idiomatic phrases across English, Portuguese, and Galician. As a result, several researchers participated in the shared task, employing diverse techniques to address this problem. These approaches included feature-based methods [64], span-based classification [152], and the use of glosses and translations [55]. Moreover, some studies explored models based on neural networks [87] and pretrained language models for idiomaticity prediction [13, 87, 139, 25], while others adopted named entity recognition techniques [97, 143].

The prediction of cross-lingual idiomaticity is an under-researched area [15]. However, the study by Swaminathan et al. [137] focused on the prediction of cross-lingual idiomaticity using multilingual pretrained language models such as multilingual BERT and XLM-R, concluding that these models are capable of learning and transferring idiomaticity information across languages. A closely related earlier

study is [39], which utilizes BERT, RoBERTa, mBERT, and RuBERT models to predict the idiomaticity of MWEs in a cross-lingual setting. The study considers previously unseen expressions by adopting a cross-lingual "zero-shot" approach, in which the model is trained on English and tested on Russian, and vice versa, making it a cross-lingual task.

2.2.4 Cross-lingual Token-level Identification of MWEs (PARSEME)

Early approaches to MWE identification used lexicon-based or rule-based methods that often relied on lists of known phrases or handcrafted rules that took advantage of fixed syntactic patterns [40]. Although these methods achieved some success with relatively high-resource languages (such as English or French) [131, 66, 37], they were not transferable to other languages because of their dependence on language-specific patterns and resources. Token-level MWE identification is important for various downstream applications, including information extraction, syntactic parsing, and machine translation [31]. Cross-lingual MWE identification is even more challenging due to linguistic diversity [51].

The PARSEME shared tasks, which are part of the PARSEME (PARSing and Multiword Expressions) initiative, have played a key role in cross-lingual MWE identification research by providing standardized datasets and evaluation metrics, as well as promoting collaboration among researchers working on different languages and methodologies [127, 114, 115, 125]. In the first edition of the PARSEME shared task [127], data from eighteen languages were released, consisting of 5.5 million tokens and 60,000 verbal MWE (VMWE) annotations. Moreover, seven teams participated in the shared tasks while five teams worked on multilingual systems. Following the success of the first shared task, the second [114] and third [115] editions of the shared

tasks were also organized, where new languages were introduced, the meaning of VMWEs was refined, and the size of the data increased. Twelve teams participated in the second edition, while seven teams participated in the third edition. Moreover, the third edition also emphasized the evaluation of unseen VMWEs and redefined the meaning of unseen VMWEs.[8] The best performing system in this shared task was MTLB-STRUCT [140], which employs mBERT with a dependency parser to identify and classify VMWEs. In addition, neural network-based approaches [156], multilingual masked language models [76, 140], and contextualized word embeddings [48] have also been studied in this third edition of the PARSEME shared task.

Another area of exploration is the handling of semi-fixed and compositional MWEs, which can appear literal in one context but idiomatic in another context [72]. This becomes more complex in a cross-lingual setting: a MWE that is idiomatic in one language could be compositional in another language. For instance, the English idiom *spill the beans*, meaning to reveal a secret, often loses its figurative meaning when translated literally into other languages such as Bangla or Hindi. Multilingual transfer learning involves adapting a model trained on one language to others with minimal resources, demonstrating potential for cross-lingual generalization, which is a field that is still under-researched [120]. Cross-lingual VMWE identification in the PARSEME 1.2 edition has been studied by Swaminathan et al. [137], where the model is trained on one language and tested on others. This study employs the MTLB-STRUCT framework and a multilingual BERT model to conduct cross-lingual experiments. The findings show that the models can learn VMWEs in a cross-lingual setting. Recent initiatives include the development of multilingual datasets [46, 51, 125] to allow cross-lingual MWE research [6, 98, 46].

[8] A VMWE from the test corpus is considered unseen if a VMWE with the same (multi-)set of lemmas is not annotated in the training or development corpus [115].

Chapter 3

Model

We employ transformer-based multilingual masked language models (MLMs) since we test and train in multiple languages in our studies. For the MLMs, we used XLM-R-large [29] and mT5-base [151] which is a multilingual model trained on T5 architecture [111]. We also used autoregressive models, GPT-4o and GPT-4o-mini [2], and considered a prompting-based approach and supervised fine-tuning.

3.1 SemEval

For SemEval 2022 task 2 subtask A [142], we apply XLM-R-large and mT5-base models for sequence classification. The shared task organizers used a multilingual BERT (mBERT) model as the baseline in the initial shared task published in 2020. Moreover, the study by Swaminathan and Cook [137] used mBERT, XLM-RoBERTa, and mDeBERTa to tackle the shared task problem. We follow the Swaminathan and Cook study and fine-tune on the training data with more powerful models to understand whether the large masked language models (e.g., XLM-R-large and mT5-base) improve the classification performance. We chose XLM-R-large to directly compare

its performance with that of its smaller version, XLM-R-base.[1] Furthermore, the reason for choosing mT5-base is that it shows better performance in various downstream multilingual NLP tasks compared to XLM-R-large [151]. We also explore large autoregressive models to see if there is any improvement over the test data. We chose two autoregressive models, GPT-4o and GPT-4o-mini, which share the same architecture but differ in model size, with GPT-4o being the larger one. This selection allows us to understand the performance differences between larger and smaller models and the capabilities of these autoregressive models compared to MLMs. We used the prompt-based and supervised fine-tuning approaches (described in Section 4.4) for autoregressive models. We chose the prompt-based approach to evaluate how these autoregressive models perform using their pretrained knowledge without additional training. The reason behind choosing the supervised fine-tuning approach is to understand how much we can improve the performance of these autoregressive models by fine-tuning with training data. For the prompt-based approach, we use prompting without training GPT-4o and GPT-4o-mini to get the model predictions. We fine-tuned both the GPT-4o and GPT-4o-mini models for the supervised fine-tuning approach and then used prompting to get the model predictions.

3.2 PARSEME

For the PARSEME 1.2 shared task [115], we first used the MTLB-STRUCT system [140], which performed the best in the shared task and was also used in the study by Swaminathan and Cook [137]. MTLB-STRUCT architecture is designed to simultaneously learn MWEs and dependency trees by constructing a dependency tree CRF network [164] while using the same pretrained model weights for both tasks. To perform the classification of MWEs based on their category in MTLB-

[1]The performance of XLM-R-base model is reported by Swaminathan and Cook et al.[137].

STRUCT, a layer containing a softmax function is added on top of the pretrained model. Moreover, MTLB-STRUCT uses Adam [73] optimizer to reduce the classification loss during training. The cross-lingual identification and classification of VMWEs have been improved using this architecture [141], where the model was trained on German and tested on English data from the PARSEME 1.1 shared task [113]. Our experimental settings include both multilingual and cross-lingual setups along with the monolingual setting using large MLM as discussed in section 4.2. We fine-tuned XLM-R-large using the MTLB-STRUCT system for the PARSEME task. To evaluate the performance of a larger MLM with a smaller MLM in terms of parameter size, we chose XLM-R-large model. In addition, we fine-tuned GPT-4o-mini to investigate whether the large autoregressive models perform better than the Swaminathan and Cook results in the cross-lingual setting. We chose GPT-4o-mini for its smaller size compared to GPT-4o and its efficiency, which allows us to evaluate the performance of a smaller autoregressive model while maintaining strong capabilities in downstream NLP tasks. Moreover, the fine-tuning of GPT-4o-mini is 12.5 and 8.3 times more cost-effective for inference and training, respectively, compared to fine-tuning of GPT-4o.[2] Our study also includes the prompt-based approach for GPT-4o, where we apply the prompting technique without training the model to obtain the model predictions.

[2]We compute these numbers based on information provided here: `https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/`

Chapter 4

Materials and Methods

We discuss our datasets, experimental setup, implementation and parameter settings, supervised fine-tuning and the prompt-based approaches, and evaluation metrics in this chapter.

4.1 Datasets

Our study focuses on cross-lingual identification and idiomaticity prediction of MWEs. Multilingual datasets are required to perform cross-lingual prediction. To address this, we used the data of 'subtask A' of "SemEval 2022 task 2: Multilingual Idiomaticity Detection and Sentence Embedding" [142] and PARSEME edition 1.2 [115] on semi-supervised identification of verbal multiword expressions.

4.1.1 SemEval

SemEval 2022 shared task 2 subtask A provides a multilingual dataset that is divided into train, dev, eval, and test sets. The train and dev sets are released mainly for

Language	MWEs	Target Sentence	Label
English	smoking gun	At least Donald Trump's "**smoking gun**" tape is simpler than Richard Nixon's.	Idiomatic
English	smoking gun	The **Smoking Gun** was a prime spot for football fans to watch the game Sunday afternoon.	Literal
Portuguese	carne branca	A procura por **carne branca** aumentou bastante.	Idiomatic
Portuguese	carne branca	LINGUADO: **carne branca** e magra, tradicionalmente preparado em filés.	Literal

Table 4.1: Examples of 'idiomatic' and 'literal' usages of MWEs for English and Portuguese from the SemEval 2022 task 2 subtask A dataset. **Bold** indicates the MWEs in the sentences.

the training of the models while the eval set is released for evaluation during the practice phase of the shared task. The test set is used for the final evaluation of the models. Among the four data splits, the train, dev, and eval sets have instances in English and Portuguese while the test set contains English (en), Portuguese (pt), and Galician (gl) instances. In our study, we used the train and dev sets to train and validate our models, while the test set was used to evaluate the performance of the models. The dataset has a total of eight columns: DataID, Language, MWE, Setting, Previous, Target, Next, and Label. We mainly used the Previous, Target, Next, and Label columns. The Previous column contains a sentence that comes before the sentence containing the target MWE while the value of the Target column is a sentence containing the target MWE. The Next column has the sentence that follows the target sentence. The values in the Label column are either '0' or '1' where 0 denotes that the target sentence contains an idiomatic MWE while 1 denotes that the target sentence has a literal MWE. The training data has $2,535$ idiomatic expression instances (approximately 56.4% of total training instances) and $1,956$ literal instances (approximately 43.6% of total training instances). Table 4.1 shows

examples of idiomatic and literal instances for the same MWEs from the dataset. The MWE *smoking gun* in the first English sentence refers to a fact that serves as conclusive evidence of an offense and is categorized as 'idiomatic' while the second sentence refers to a place name that is categorized as 'literal'. Similarly for the first sentence of Portuguese, *carne branca* is used as an idiomatic expression and the second sentence shows the literal meaning for the same MWE.

Language	**Train**	**Dev**	**Test**
English	3,327	466	916
Portuguese	1,164	273	713
Galician	0	0	713
Total	**4,491**	**739**	**2,342**

Table 4.2: Number of instances across the train, dev, and test splits for SemEval 2022 task 2 subtask A.

In this thesis, only the "zero-shot" setting from the shared task is considered. In this setting, the training MWEs are completely different from the MWEs in the dev and test sets. In this study, we only consider cross-lingual experiments in which a model is evaluated using expressions in a language which was not seen during training. We evaluate the models on the test dataset focusing on the results for languages that were not seen during training (e.g., when training on English, we consider the results for Portuguese and Galician for analysis). Since our experiments are cross-lingual (i.e., the training and testing languages are different), they are also "zero-shot" experiments in nature. We present the statistics for the SemEval 2022 task 2 subtask A dataset in Table 4.2.

4.1.2 PARSEME

The PARSEME 1.2 shared task [115] provides a multilingual dataset which includes 14 languages ranging from low- to high-resource languages. This dataset contains sentences with token-level annotation for verbal MWEs (VMWEs). VMWEs are MWEs for which the head of the expression is a verb. In this dataset, the tokens are annotated into nine categories of MWEs for 14 languages which are Basque, Brazilian Portuguese, Chinese, French, German, Greek, Hebrew, Hindi, Italian, Irish, Polish, Romanian, Swedish, and Turkish. The dataset is divided into train, dev, and test sets. The train and dev sets are provided for training and validating the models while the test set is used to evaluate the models. In this shared task, the models are expected to correctly predict the VMWEs of the test set.

#	**Input**	**Gold Standard**
1	Úsáideann	*
2	busanna	*
3	na	*
4	páirce	*
5	ola	*
6	ráibe	*
7	agus	*
8	tá	*
9	achan	*
10	iarracht	1:LVC.full
11	déanta	1
12	an	*
13	saol	*
14	glas	*
15	a	*
16	chur	2:LVC.cause
17	chun	2
18	cinn	2
19	.	*

Table 4.3: Example of an instance for the PARSEME task dataset in Irish.

We present the token-level annotation for an Irish sentence from this dataset in

Table 4.3, where '*' denotes that the input token has not been identified as part of a VMWE. In the example, the gold standard indicates that *iarracht déanta* is a full light verb construction (LVC.full) and *chur chun cinn* is a causative light verb construction (LVC.cause).

Language	**Train**	**Dev**	**Test**
German (de)	6,568	602	–
Greek (el)	17,733	909	–
Basque (eu)	4,440	1,418	–
French (fr)	14,377	1,573	–
Irish (ga)	257	322	1,121
Hebrew (he)	14,152	1,254	–
Hindi (hi)	282	289	1,113
Italian (it)	10,641	1,202	–
Polish (pl)	17,731	1,425	–
Brazilian Portuguese (pt)	23,905	1,976	–
Romanian (to)	10,920	7,714	–
Swedish (sv)	1,605	596	–
Turkish (tr)	17,945	1,062	–
Chinese (zh)	35,326	1,141	–
All	175,882	21,483	2,234

Table 4.4: Number of training, development, and testing examples we used in our study from the PARSEME 1.2 dataset.

Chinese, Irish, and Swedish were newly introduced or changed significantly in PARSEME 1.2 compared to the previous version of the PARSEME shared task. PARSEME 1.2 mainly emphasized evaluating on unseen VMWEs by redefining the definition of unseen VMWEs as the collection of lemmas that are not annotated in both the train and dev datasets, as opposed to only in the train dataset as in the previous PARSEME edition.[1] In this study, we evaluate our model only on Irish and Hindi test data. The reason behind choosing Irish is due to it having the lowest performance in the "Heldout" setting from the Swaminathan and Cook baseline [137], while Hindi was chosen because it has the least amount of training samples among

[1] A **Lemma** is the base or dictionary form of a word that is used to represent all its inflected variants in linguistic analysis.

the other languages in the dataset. We exclude other languages because of the high computational cost of fine-tuning the large autoregressive model.[2] Table 4.4 shows the number of training, development, and testing examples that we used in our study from the PARSEME 1.2 dataset. In terms of unseen VMWEs, the Irish and Hindi test data have 353 and 370 unseen VMWEs. Although the PARSEME data for each language is divided into training, development, and test sets, we only used the training and development sets for all languages other than Irish and Hindi in our experiments.

4.2 Experimental Setup

Our experiments were designed following the experimental settings of Swaminathan and Cook [137]. For this thesis, we mainly considered the following four experimental settings to perform cross-lingual analysis on SemEval 2022 shared task 2 Subtask A:

1. **no train:** No model is trained for this experiment. This experiment is designed only for autoregressive models and uses the prompt-based approach.

2. **en:** Two MLMs (XLM-R-large and mT5-base) and two autoregressive models (GPT-4o and GPT-4o-mini) are trained on only English training data.

3. **pt:** The same as en, but here only Portuguese training data is used to train all four language models.

4. **en+pt:** The same as the previous two approaches but here both English and Portuguese training data are utilized to train all four language models.

[2]For example, supervised fine-tuning of the "All" experiment setting (discussed in section 4.2) costs roughly 525 US dollars for each language.

The first setting is mainly considered to evaluate the capabilities of large autoregressive models without training. Since the models are not trained, evaluating on expressions in any language can be viewed as a cross-lingual task because the model was not trained on the training instances in the test language. This experiment setting helps us to answer the first sub-question of the second research question.[3] In the second setting, we only train on English. Therefore, when we test on Portuguese or Galician, this becomes a cross-lingual task since both Portuguese and Galician were unseen during training. Similarly, for the third setting, we only use Portuguese for training. This setting then becomes a cross-lingual task when we test on English or Galician since both were unseen during training. Lastly, we combine the English and Portuguese training data for training in the fourth experiment setting. This becomes a cross-lingual task when we test on Galician data. The second, third, and fourth experiment settings help us to answer the first[4] and second[5] research questions along with the second sub-question[6] of the second research question.

We also follow experimental settings similar to Swaminathan and Cook [137] for the PARSEME shared task. Our motivation behind designing experimental settings is to understand the performance of monolingual, multilingual, and cross-lingual approaches for all languages. As a result, we considered three experimental settings to answer the research questions. However, due to limited computational resources and the high cost of fine-tuning large autoregressive models such as GPT-4o-mini, we considered our three experimental settings for only two languages (Irish and Hindi). We discuss our experimental settings below:

1. **Mono:** In this setting, we train on the training set of a particular language

[3] *Does the prompting-based approach of the large autoregressive models outperform the approach of Swaminathan and Cook [137] in cross-lingual settings?*

[4] *Do larger MLMs outperform the approach of Swaminathan and Cook in cross-lingual settings?*

[5] *Do larger autoregressive models outperform the approach of Swaminathan and Cook in cross-lingual settings?*

[6] *Does supervised fine-tuning of larger autoregressive models outperform the approach of Swaminathan and Cook in cross-lingual settings?*

and test on the same language using all language models. Models are trained for each language separately.

2. **All:** In this setting, we aim to analyze the performance of the multilingual behavior of the models by incorporating additional training data from other languages and training on all languages instead of just a specific language as in the Mono setting. However, due to computational resource constraints, we excluded the Hindi language for the Irish "All" experimental setting and the Irish language for the Hindi "All" experimental setting. I.e., we use the same models that were trained for the Hindi and Irish "Heldout" settings (described below). This experiment setting is therefore slightly different from the Swaminathan and Cook "all" setting for the PARSEME task.

3. **Heldout:** This setting is mainly designed to analyze the cross-lingual capabilities of the language models. We trained the model on all of the languages except the target language and tested the model on the target language. Models are again trained for each language separately. I.e, we train one model on all languages except Hindi and evaluate on Hindi and then separately train another model on all languages except Irish and evaluate on Irish.

4.3 Implementation and Parameter Settings

For fine-tuning large MLMs, we used Huggingface's transformers [148] implementations and publicly available weights of XLM-RoBERTa (XLM-R) and multilingual T5 (mT5) models. In particular, we used the large architecture (*XLM-R-large*) of XLM-R and the base architecture (*mT5-base*) of mT5. XLM-R is a pre-trained model using the XLM approach [30] on 2.5TB of filtered CommonCrawl data containing 100 languages. XLM-R-large has 24 layers, 1024 hidden states, and 16 attention

heads of the transformers architecture [29] while the dimension of the feed-forward network is 4096. Moreover, the model has 550 million parameters and the vocabulary size is $250k$. mT5 uses the T5 architecture trained on the mC4 corpus and covers 101 languages ranging from low- to high-resource languages [151]. mT5-base has 12 layers, 768 hidden states, and 12 attention heads of the transformers architecture [29] while the dimension of the feed-forward network is 2048. Moreover, mT5 has 580 million parameters and the vocabulary size is $250k$, and it uses SentencePiece tokenizer.

For the prompting-based approach and supervised fine-tuning of large autoregressive models, we used the Python implementation of the OpenAI API library.[7] We used GPT-4o (version gpt-4o-2024-08-06) and GPT-4o-mini (version gpt-4o-mini-2024-07-18) [2] in this thesis. Most of the technical details of GPT-4o and GPT-4o-mini such as model architecture, number of parameters, training data, etc. are unknown because the models are closed source. However, the size of the context window and the maximum number of output tokens are 128,000 and 16,384 tokens respectively for both models, suggesting that the models are capable of understanding long input contents. The main difference between the two models is their size and computational cost. GPT-4o is larger in size and offers better performance on complex tasks compared to GPT-4o-mini [107].

We fine-tuned XLM-R-large, mT5-base, GPT-4o, and GPT-4o-mini for the SemEval task and XLM-R-large and GPT-4o-mini for the PARSEME task. We used MTLB-STRUCT [140] to fine-tune XLM-R-large for the PARSEME task. Furthermore, we used GPT-4o for the prompt-based approach for both the SemEval and PARSEME tasks.

We used a learning rate of 3×10^{-5} for optimizer AdamW [85], a maximum sequence

[7] `https://github.com/openai/openai-python`

length of 256, and a batch size of 16 for fine-tuning MLMs for both tasks and trained up to 10 epochs.[8] For the prompt-based approach, we set the *temperature* to 0, *top_p* to 0.95, and instructed the model to generate up to 16 tokens. For supervised fine-tuning of GPT-4o-mini, we used a batch size of 2, *learning rate multiplier* of 0.5 and trained up to 3 epochs for the SemEval task.[9] For the PARSEME task, we train GPT-4o-mini only using the same parameters, however, we reduced the number of epochs for training for the "All" and "Heldout" experimental settings to 2 due to limited computational resources.

To obtain results over the SemEval test data, we uploaded the models' predictions to the competition website[10] since the test data labels were not publicly available when we conducted the experiments.

4.4 Supervised Fine-tuning and Prompt-based Approach for Autoregressive Models

We constructed an instruction-following dataset to fine-tune the autoregressive models. The motivation for constructing an instruction-following dataset is to use a standard approach to fine-tune an LLM [162] to enhance the model's generalizability and guide the LLM to follow user instructions. We provide an example of instruction-following data constructed using 'User' prompt in Table 4.5 for the SemEval and PARSEME tasks.

For the prompt-based approach, we manually carefully designed prompts to instruct autoregressive models to perform both tasks. Similarly, we also designed the prompt

[8]Learning rate controls how much the model updates its parameters (like weights) in response to the error it sees after each step of training.

[9]Learning rate multiplier is a factor applied to the base learning rate to scale it differently for training stages.

[10]`https://codalab.lisn.upsaclay.fr/competitions/8121`

to instruct supervised fine-tuned autoregressive models. Our prompt engineering approach is motivated by our experimental findings across trial data. For this experiment, we use the system and user prompts in Table 4.5.

4.5 Evaluation Metrics

The classes are imbalanced and the majority of the data belongs to the idiomatic class for the SemEval task. Moreover, around 74% of training instances are in English, and the remaining 26% of training instances belong to Portuguese. Therefore, we follow the official evaluation metrics of the shared tasks and evaluate our model predictions using the macro-average F1 score, which is the macro-averaged harmonic mean of the precision and recall for individual classes.[11]

We also noticed similar class imbalances in the PARSEME task. Therefore, we used the official evaluation metrics of the shared task: global token-based F1 score, global MWE-based F1 score, and unseen MWE-based F1 score. The precision and recall of the predicted VMWE boundaries are measured in the global token-based evaluation. The precision and recall of complete VMWEs, including their type (e.g., VID, IAV), are measured in the global MWE-based evaluation. The unseen MWE-based evaluation considers only VMWEs that are not seen in training or development data. Note that in the case of the 'Heldout' experimental setting, all the test expressions are not observed during training.

For the SemEval task, we compare our results against the most frequent class (MFC) baseline and the Swaminathan and Cook [137] results. The MFC baseline classifies every instance as the MFC, in this case idiomatic. For the PARSEME task, we only compare against the Swaminathan and Cook results.

[11]Macro refers to the method of averaging F1 scores unweighted across all classes.

Role	Prompt
	SemEval task
System	You are an expert in identifying the idiomatic expressions by analyzing the given sentence.
User	Analyze the given sentence and determine whether it contains an idiomatic expression (a phrase or saying where the meaning cannot be inferred directly from the individual words) or a literal expression (where the words convey their standard meaning). Your answer should be either 'literal' or 'idiomatic'. Provide only class as your response. sentence: {sentence} class: {class_label}
	PARSEME task
System	You are an expert in identifying verbal multi-word expressions (VMWE) from the given sequence of words.
User	Analyze the given sequence of words and determine VMWE for each word. The VMWE categories are 'LVC.full', 'LVC.cause', 'VID', 'IRV', 'VPC.full', 'VPC.semi', 'MVC', and 'IAV'. Follow the rules described below for the entire input. 1. Write the appropriate VMWE category if the current line contains the first lexicalized component of the VMWE in the sentence, the VMWE code consists of a VMWE identifier followed by a colon (:) and a VMWE category label (for example: 1:VID). VMWE identifiers are integers starting from 1 for each new sentence and increasing by 1 for each new VMWE. 2. If the current line contains a lexicalized component of the VMWE which is not the first one in the sentence, the VMWE code contains the VMWE identifier only, as described above, and no VMWE category label. 3. Write a star (*) for the word if the word in the current line is not part of a VMWE, or if the current line describes a multiword token (e.g., 2-3, don't). 4. Write an underscore (_) for the word if this information is underspecified. The input data is provided in tsv format with the following columns: ID\tWord\tPOS {input_data}

Table 4.5: Prompts used with the LLMs to get the predictions for both the prompt-based approach and supervised fine-tuning.

Chapter 5

Results

In this chapter, we present the results of the SemEval task in Section 5.1 and the PARSEME task in Section 5.2.

5.1 SemEval

We present the results of the SemEval task in Table 5.1, where the values are presented as macro averaged F1 scores and rounded to three decimal places. We also present the most frequent class baseline along with the best results of individual language settings of Swaminathan and Cook [137] as another baseline. Our study focuses mainly on cross-lingual settings, where the models are trained on one language and tested on different languages, e.g., we trained a model using English data and tested the model using Portuguese and Galician data. This section first discusses the performance of larger masked language models (MLMs) e.g., XLM-R-large and mT5-base compared to Swaminathan and Cook [137]. Then we discuss the performance of autoregressive models such as GPT-4o and GPT-4o-mini. Finally, we conclude the section by answering the research questions.

Model	Train	Test			
		en	**pt**	**gl**	**All**
XLM-R-large	en	0.703	0.523	0.321	0.532
	pt	0.625	0.631	0.444	0.572
	en + pt	0.724	0.679	0.469	0.648
mT5-base	en	0.681	0.541	0.392	0.551
	pt	0.367	0.391	0.434	0.400
	en + pt	0.348	0.393	0.458	0.394
GPT-4o	no train	0.665	0.598	0.502	0.602
	en	0.788	**0.653**	**0.588**	0.696
	pt	**0.779**	0.651	**0.659**	0.714
	en + pt	0.802	0.664	**0.657**	0.743
GPT-4o-mini	no train	0.622	0.499	0.285	0.481
	en	0.780	**0.611**	0.380	0.609
	pt	**0.724**	0.632	**0.579**	0.658
	en + pt	0.785	0.704	0.543	0.693
Baseline [137]	en	0.717	0.590	0.420	–
	pt	0.582	0.578	0.499	–
	en + pt	0.720	0.662	0.550	–
MFC Baseline	–	0.345	0.391	0.434	0.389

Table 5.1: Macro-average F1 score for the SemEval task for each model, training and testing on the indicated language(s). **Bold** indicates better performance than the Swaminathan and Cook baseline for the cross-lingual setting.

Training on English and testing on Portuguese, the performance of XLM-R-large (0.523) is better than the most frequent class (MFC) baseline. However, the performance of XLM-R-large is lower than the Swaminathan and Cook baseline (0.590) for the same experiment setting by a margin of 0.067. When trained on English and tested on Galician, the performance of XLM-R-large (0.321) is lower than both the MFC baseline (0.434) and the Swaminathan and Cook baseline (0.420). Similarly, when trained on English using mT5-base and tested on Portuguese, the performance

(0.541) is better than the MFC baseline (0.391) but lower than the Swaminathan and Cook baseline (0.590). For the same training setting when tested on Galician, the performance (0.392) is lower than both the MFC baseline (0.434) and the Swaminathan and Cook baseline (0.420) by a margin of 0.042 and 0.028 respectively. Moreover, when trained on English and tested on Portuguese and Galician, the performance of mT5-base is better than the performance of XLM-R-large.

When trained on Portuguese using XLM-R-large and tested on English, our performance (0.625) is better than both the MFC baseline (0.345) and the Swaminathan and Cook baseline (0.582). For the same trained model when we tested on Galician, the performance of XLM-R-large (0.444) is better than the MFC baseline (0.434), however, it is lower than the Swaminathan and Cook baseline (0.499) by a margin of 0.055. When trained on Portuguese and tested on English, the performance of mT5-base (0.367) is better than the MFC baseline (0.345) by a small margin of 0.022, however, it is lower than the Swaminathan and Cook baseline (0.582) for the same experiment setting. Moreover, the performance difference between both models (XLM-R-large and mT5-base) is quite large with XLM-R-large outperforming mT5-base by 0.258 when tested on English. This could be due to the massive amount of data present in English in the mT5-base model [151] and fine-tuning with Portuguese data does not help the model in learning compared to fine-tuning with English data. When trained on Portuguese and tested on Galician, the performance of mT5-base (0.434) is the same as the MFC baseline. We investigated the output of mT5-base for Galician to understand whether the model predicted all the test data into the most frequent class, which might be expected because it achieves the same F1-score as the MFC baseline, however, we found that the model in fact predicts both classes.

For training on English and Portuguese and testing on Galician, the performance of XLM-R-large (0.469) is better than the MFC baseline (0.434), however, it is lower than the Swaminathan and Cook baseline (0.550). For the same experiment setting

using mT5-base, we observe the performance (0.458) is better than the MFC baseline by a small margin of 0.024 while it is lower than the Swaminathan and Cook baseline.

Our experimental results using larger MLMs show that training on Portuguese and testing on Galician performs better than training on English and testing on Galician. This could be due to both Portuguese and Galician being part of the Romance language family. Moreover, the performance of mT5-base when trained on English and tested on Portuguese (0.541) is better than the performance of mT5-base when trained on Portuguese and tested on Portuguese (0.391). This could be due to the small amount of data present in mT5-base for Portuguese compared to English.

For the prompting-based approach (no train), we only compare our results with the MFC baseline because this is the only baseline considered which also does not use training data. For the prompting-based approach tested on English, the performance of GPT-4o (0.665) and GPT-4o-mini (0.622) is better than the MFC baseline (0.345). When tested on Portuguese, the performance of GPT-4o (0.598) and GPT-4o-mini (0.499) is again better than the MFC baseline (0.391). When tested on Galician for the prompting-based setting, the performance of GPT-4o (0.502) is better than the MFC baseline (0.434), however, the performance of GPT-4o-mini (0.285) is lower than the MFC baseline. One possible reason for the low performance of GPT-4o-mini could be that it is smaller in model size and has a smaller amount of training data compared to GPT-4o.

Training on English and testing on Portuguese, supervised fine-tuning of GPT-4o and GPT-4o-mini outperforms both baselines, achieving the best result using GPT-4o (0.653) in any cross-lingual setting for Portuguese. When trained on English and tested on Galician, the performance of GPT-4o (0.588) is better than both the MFC baseline (0.434) and the Swaminathan and Cook baseline (0.420), while the performance of GPT-4o-mini (0.380) is lower than both baselines. When trained on

Portuguese and tested on English, the performance of GPT-4o (0.779) and GPT-4o-mini (0.724) is better than both the MFC baseline (0.345) and the Swaminathan and Cook baseline (0.582), while GPT-4o provides the best result (0.779) for English compared to any cross-lingual settings. When trained on Portuguese and tested on Galician, the performance of GPT-4o (0.659) and GPT-4o-mini (0.579) is better than both the MFC baseline (0.434) and the Swaminathan and Cook baseline (0.499), while GPT-4o provides the best result (0.659) on Galician in any experiment settings when trained on Portuguese. Training on both English and Portuguese and testing on Galician, the performance of GPT-4o (0.657) is better than both MFC baseline (0.434) and the Swaminathan and Cook baseline (0.550) while the performance of GPT-4o-mini (0.543) is only better than MFC baseline and the performance difference between GPT-4o-mini and the Swaminathan and Cook baseline is quite small ($\sim$0.007).

Overall, our findings on cross-lingual settings indicate that both large MLMs and large autoregressive models are able to learn idiomaticity information that is not language-specific. However, larger MLMs could not perform better than the Swaminathan and Cook baseline for Portuguese and Galician when trained on English, but perform better for English when trained on Portuguese using XLM-R-large, indicating that the larger MLMs could not consistently perform better than the approach of Swaminathan and Cook that answers our first research question (RQ).[1] In addition, the performance of GPT-4o for the prompting-based approach is better than the Swaminathan and Cook baseline, however, the Swaminathan and Cook baseline is better than our prompting-based approach when trained on 'en + pt' and tested on Galician. Moreover, the performance of GPT-4o-mini for the prompting-based approach is lower than the Swaminathan and Cook baseline, indicating that the prompting-based approach often performs better than the approach of Swaminathan

[1] *Do larger MLMs outperform the approach of Swaminathan and Cook [137] in cross-lingual settings?*

and Cook, answering our first subquestion – *Does the prompting-based approach of the large autoregressive models outperform the approach of Swaminathan and Cook in cross-lingual settings?* – of the second RQ. The performance of supervised fine-tuned GPT-4o is better than the Swaminathan and Cook baseline for cross-lingual settings. Moreover, the performance of supervised fine-tuned GPT-4o-mini is better than the Swaminathan and Cook baseline except for 'en' and 'en + pt' experiment settings when tested on Galician, indicating that supervised fine-tuning of both autoregressive models performs better than the approach of Swaminathan and Cook in most cases that answers our second subquestion – *Does supervised fine-tuning of larger autoregressive models outperform the approach of Swaminathan and Cook in cross-lingual settings?* – of the second RQ. Moreover, our findings indicate that autoregressive models often perform better than the approach of Swaminathan and Cook, which answers the second research question – *Do larger autoregressive models outperform the approach of Swaminathan and Cook in cross-lingual settings?*

The results also emphasize the challenges of cross-lingual transfer, particularly for underrepresented languages like Galician. Our findings can be useful, particularly in those languages where resources are very limited. Our cross-lingual setup can be useful to learn idiomaticity information from resource-rich languages like English and Portuguese and test it on examples from low-resource languages like Galician. Moreover, our prompting-based approach which does not rely on training data shows that large autoregressive models like GPT-4o can be applied for low-resource languages where training data is not available. Although GPT-4o demonstrates superior performance across different experimental setups, GPT-4o-mini offers a good trade-off between performance and efficiency where computational resources are limited.

5.2 PARSEME

Table 5.2 shows results for the PARSEME task in terms of F1 score, which has been rounded to three decimal places. For each language and setting, the table compares our fine-tuned model (GPT-4o-mini) with the Swaminathan and Cook baseline [137]. In our monolingual ("Mono") setting, we fine-tuned and tested GPT-4o-mini in the same language. In the "All" experiment setting, the model is trained on all the training examples for 13 languages out of 14 languages.[2] In the "Heldout" setting, we trained the model in all the languages except the target language, e.g., the Irish "Heldout" model is trained in all languages other than Irish. We present results over unseen instances based on the monolingual training and development data for each setting when computing the unseen MWE-based F1 score ("Unseen" in Table 5.2). This enables us to understand how well the model is performing on the MWEs that are not seen during the training of the model. However, note that all test instances are in fact unseen during training in the "Heldout" experimental setup. Nevertheless, we maintain the same strategy to calculate unseen MWEs for the "Heldout" setting to enable fair comparisons across the different experimental settings. In this section, we first discuss the performance in the "Mono" experiment setting followed by the "All" and "Heldout" settings. Then we answer the research questions by comparing our results with the Swaminathan and Cook baseline. Finally, we conclude the section with a discussion of performance on individual VMWE categories for the "Heldout" setting.

In the "Mono" experimental setting for Irish, the performance of fine-tuned GPT-4o-mini for MWE, Token, and Unseen is 0.000, 0.015, and 0.000, respectively, indicating the model could not predict any MWEs correctly. We further investigated the chal-

[2]Due to computational resource constraints, we excluded the Hindi language for the Irish "All" experimental setting and Irish language for the Hindi "All" experimental setting. I.e., we use the same models that were trained for the Hindi and Irish "Heldout" settings.

		GPT-4o-mini			Swaminathan and Cook Baseline [137]		
Lang	**Setting**	**MWE**	**Token**	**Unseen**	**MWE**	**Token**	**Unseen**
ga	**Mono**	0.000	0.015	0.000	0.311	0.465	0.210
	All	**0.425**	0.455	**0.418**	0.422	0.483	0.301
	Heldout	**0.349**	**0.387**	**0.352**	0.111	0.133	0.069
hi	**Mono**	0.598	0.633	0.501	0.729	0.785	0.504
	All	0.673	0.695	**0.570**	0.759	0.796	0.549
	Heldout	**0.509**	**0.567**	**0.443**	0.376	0.452	0.278

Table 5.2: Performance (F1 score) on the PARSEME task on Irish (ga) and Hindi (hi) for monolingual ("Mono"), "All", and "Heldout" experimental settings. **Bold** indicates better performance over the Swaminathan and Cook baseline in each evaluation category.

lenges of correctly predicting MWEs and trained the model again with 5 epochs.[3] We found that training the model with more epochs improves the performance over the original training setting.[4] This indicates that training for more epochs when the number of training samples is small can improve the performance. In the "All" setting for Irish, our model performs better than the Swaminathan and Cook baseline for MWE by a very small margin (0.003) and unseen by a margin of 0.117, but the performance for token (0.455) is lower than the Swaminathan and Cook baseline (0.483). For Hindi, although the performance of our model in the "Mono" and "All" experimental settings was not better than the Swaminathan and Cook baseline for MWE and Token, the performance difference for unseen is quite small with the Swaminathan and Cook baseline outperforming our model by 0.003 for the "Mono" setting, while our model performs better on the "All" setting for the unseen evaluation measure by a margin 0.021. Moreover, training using examples from 13 languages improves the model performance for MWE, Token, and Unseen. This suggests that using information from different languages can be more beneficial than

[3]We mainly train 3 epochs for the "Mono" experimental setting.

[4]The F1 score for MWE, Token, and Unseen for 5 training epochs are 0.222, 0.323, and 0.195 respectively.

using only one language.

In the "Heldout" setting for both languages, our supervised fine-tuned models perform better than the Swaminathan and Cook baseline in all three evaluation measures. This finding shows that fine-tuned larger autoregressive models such as GPT-4o-mini are better at identifying MWEs than the Swaminathan and Cook baseline. The answers to the second research question – *Do larger autoregressive models outperform the approach of Swaminathan and Cook in cross-lingual settings?* – as well as the second subquestion – *Does supervised fine-tuning of larger autoregressive models outperform the approach of Swaminathan and Cook in cross-lingual settings?* – of the second research question are therefore "yes". To answer the first subquestion – *Does the prompting-based approach of the large autoregressive models outperform the approach of Swaminathan and Cook in cross-lingual settings?* – of the second question, we ran the experiment using the prompting-based approach, however, GPT-4o-mini was unable to correctly predict a single MWE over the entire test set of both languages, indicating that the answer to this research question is "no". To answer the first question – "*Do larger MLMs outperform the approach of Swaminathan and Cook [137] in cross-lingual settings?*", we also fine-tuned the larger MLM (XLM-R-large) and obtained an F1 score of 0.001 for MWE-based in "Heldout" experiment setting for Hindi, which is very low compared to the Swaminathan and Cook baseline (0.376), indicating that the answer to this research question is "no".

To better understand the performance on individual VMWE categories, we present the results of each VMWE category for Irish in Table 5.3, where we again rounded the results to three decimal places. In the PARSEME 1.2 [115] test data, Irish does not have instances for multi-verb constructions (MVC) and inherently clitic verbs (LS.ICV). In the "Heldout" setting for Irish, the best results are for full light verb constructions (LVC.full) with an F1 score of 0.434, while the performance for causative light verb constructions (LVC.cause), verbal idioms (VID), and full verb-

Category	Frequency	Mono	All	Heldout
IRV	6	0.000	0.000	0.000
LVC.cause	74	0.000	0.244	0.078
LVC.full	137	0.000	0.563	0.434
VID	69	0.000	0.095	0.097
VPC.full	20	0.000	0.237	0.262
VPC.semi	13	0.000	0.000	0.000
MVC	0	0.000	0.000	0.000
IAV	117	0.000	0.164	0.000
LS.ICV	0	0.000	0.000	0.000

Table 5.3: Frequency of each VMWE category for Irish along with the MWE-based F1 score for each experiment setting.

particle constructions (VPC.full) are 0.078, 0.097, and 0.262, respectively. However, despite our model outperforming the Swaminathan and Cook baseline in the "Heldout" setting, our model could not correctly predict any inherently reflexive verbs (IRV), semi verb-particle constructions (VPC.semi), or inherently adpositional verbs (IAV). Moreover, some of the IAV MWEs were correctly predicted in the "All" setting. This could be due to Italian training data containing IAV examples (324) that improve overall performance.

The results for each VMWE category for Hindi are shown in Table 5.4. Hindi test data do not have instances for IRV, VPC.full, VPC.semi, IAV, and LS.ICV. As a result, the model performance has been impacted due to these missing VMWE categories. For example, although the IRV category was not present in Hindi test data, IRV was predicted in the "Heldout" experiment settings. The best result for the "Heldout" setting is for LVC.full, while the performance on LVC.cause and VID are 0.189 and 0.113 respectively. Moreover, LVC.cause is only correctly predicted in the "Heldout" experimental setting. Although the best result in the "All" experimental setting is for MVC, the model could not correctly predict any MVCs in the "Heldout"

Category	Frequency	Mono	All	Heldout
IRV	0	0.000	0.000	0.000
LVC.cause	23	0.000	0.000	0.189
LVC.full	406	0.617	0.667	0.639
VID	39	0.000	0.196	0.113
VPC.full	0	0.000	0.000	0.000
VPC.semi	0	0.000	0.000	0.000
MVC	205	0.463	0.721	0.000
IAV	0	0.000	0.000	0.000
LS.ICV	0	0.000	0.000	0.000

Table 5.4: Frequency of each VMWE category for Hindi along with the MWE-based F1 score for each experiment setting.

setting. This could be because MVC is not a common type of MWE in the other languages. Although the Chinese training data contain a large amount of MVC examples, it does not appear to help the model to learn the MVC category for the "Heldout" setting.

Chapter 6

Conclusions

Multiword expressions (MWEs) pose a substantial challenge in NLP due to their idiomatic nature, making them difficult to interpret based solely on the meanings of their individual component words. Identifying MWEs is important to improve the performance of downstream NLP tasks such as machine translation, where failing to recognize idiomatic expressions can lead to literal and incorrect translations, and sentiment analysis, where failing to capture idiomatic expressions can lead to interpreting sentiment incorrectly. Although MWE identification in monolingual contexts has advanced significantly, limited work has been done on cross-lingual MWE identification, particularly in the "zero-shot" setting [39, 137], in which models are applied to languages that were not seen in training. In this thesis, we studied cross-lingual settings for the SemEval 2022 Task 2 Subtask A and PARSEME 1.2 shared tasks. Additionally, we investigated whether using additional training data from other languages could improve model performance.

The first research question that we investigated is *Do larger masked language models (MLMs) outperform the approach of Swaminathan and Cook [137] in cross-lingual settings?* Our findings in Table 5.1 indicate that larger MLMs for the SemEval data

could not perform consistently better than the approach of Swaminathan and Cook, especially on Portuguese and Galician when trained on English, but performed better on English when trained on Portuguese using XLM-R-large. Moreover, our approach using mT5-base could not perform better than the approach of Swaminathan and Cook. Similarly, larger MLMs could not perform better on the PARSEME task. Then, we investigated the second research question that is *Do larger autoregressive models outperform the approach of Swaminathan and Cook in cross-lingual settings?* To address this research question, we also explored two more research questions, which are "*Does the prompting-based approach of the large autoregressive models outperform the approach of Swaminathan and Cook in cross-lingual settings?*" and "*Does supervised fine-tuning of larger autoregressive models outperform the approach of Swaminathan and Cook in cross-lingual settings?*".

Our findings show that the prompting-based approach often performs better than the approach of Swaminathan and Cook on the SemEval task using GPT-4o, particularly on English and Portuguese, but not using GPT-4o-mini. However, supervised fine-tuning of both autoregressive models performs better than the approach of Swaminathan and Cook in most cases, but not using GPT-4o-mini when trained on English and "English + Portuguese" and tested on Galician. Moreover, the prompting-based approach for the PARSEME task could not perform better than the approach of Swaminathan and Cook. However, supervised fine-tuning of large autoregressive model performed better than the approach of Swaminathan and Cook in cross-lingual experiments.

Future research can explore several directions, including hyperparameter optimization, extending supervised fine-tuning of autoregressive models to cover all languages in the PARSEME 1.2 task, investigating cross-lingual transfer learning techniques to enhance MWE identification across diverse linguistic settings, investigating multimodal idiomaticity prediction in multilingual settings, and leveraging large open-

source autoregressive models for improved generalization.

Hyperparameter optimization could be a future direction for the PARSEME task, including the number of epochs, batch size, and learning rate. For example, we noticed that an increasing number of training epochs performs better in the "Mono" setting for Irish in the PARSEME task.

We intend to test our models in the cross-lingual setting on the remaining languages of PARSEME edition 1.2, as well as the recently released PARSEME edition 1.3 shared task [125] that presents several changes and improvements over the PARSEME edition 1.2 shared task. The datasets for Chinese, Greek, and Swedish have been expanded with additional instances, and new languages such as Arabic and Serbian have been introduced. Moreover, PARSEME 1.3 improves the dataset by including sentences with previously excluded MWE types (e.g., inherently adpositional verbs (IAVs)) in languages such as Croatian and Romanian.

Another prominent research direction could be multimodal idiomaticity prediction in the multilingual setting. Recently, Pickard et al. [105] proposed a shared task on advancing multimodal idiomaticity representation in multiple languages, which could be utilized for cross-lingual multimodal idiomaticity prediction employing both textual and visual features.

Although our study includes large closed-source autoregressive language models (such as GPT-4o and GPT-4o-mini), we plan to extend our model choices to open-source models (such as Llama 3, Mistral Large Instruct 2407, etc.) to understand the cross-lingual capabilities of these models. These models have shown state-of-the-art performance on various downstream multilingual tasks, including reasoning and machine translation [49]. As a result, we would like to evaluate these powerful for MWE identification. Furthermore, the study by Brown et al. [17] demonstrated that language models are few-shot learners. Therefore, we plan to use in-context

few-shot techniques using large autoregressive models for the MWE identification and idiomaticity prediction tasks. This approach aims to improve model adaptability while significantly reducing the computational costs associated with supervised fine-tuning [83], making it a more efficient and scalable solution.

Bibliography

[1] Vahid Nejad Mahmood Abadi and Fahimeh Ghasemian, *Enhancing persian text summarization through a three-phase fine-tuning and reinforcement learning approach with the mt5 transformer model*, Scientific Reports **15** (2025), no. 1, 80.

[2] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al., *Gpt-4 technical report*, arXiv preprint arXiv:2303.08774 (2023).

[3] A Anaby-Tavor, B Carmeli, E Goldbraich, A Kantor, G Kour, S Shlomov, N Tepper, and N Zwerdling, *Not enough data? deep learning to the rescue!*, arXiv preprint arXiv:1911.03118 (2019).

[4] Darko Anicic, Paul Fodor, Sebastian Rudolph, Roland Stühmer, Nenad Stojanovic, and Rudi Studer, *A rule-based language for complex event processing and reasoning*, Web Reasoning and Rule Systems: Fourth International Conference, RR 2010, Bressanone/Brixen, Italy, September 22-24, 2010. Proceedings 4, Springer, 2010, pp. 42–57.

[5] Mikel Artetxe, Gorka Labaka, and Eneko Agirre, *Translation artifacts in cross-lingual transfer learning*, Proceedings of the 2020 Conference on Empirical

Methods in Natural Language Processing (EMNLP) (Online) (Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, eds.), Association for Computational Linguistics, November 2020, pp. 7674–7684.

[6] Andrei-Marius Avram, Verginica Barbu Mititelu, Vasile Păiș, Dumitru-Clementin Cercel, and Ștefan Trăușan-Matu, *Multilingual multiword expression identification using lateral inhibition and domain adaptation*, Mathematics **11** (2023), no. 11, 2548.

[7] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio, *Neural machine translation by jointly learning to align and translate*, arXiv preprint arXiv:1409.0473 (2014).

[8] Timothy Baldwin, Colin Bannard, Takaaki Tanaka, and Dominic Widdows, *An empirical model of multiword expression decomposability*, Proceedings of the ACL 2003 workshop on Multiword expressions: analysis, acquisition and treatment, 2003, pp. 89–96.

[9] Timothy Baldwin and Su Nam Kim, *Multiword expressions*, Handbook of Natural Language Processing (Nitin Indurkhya and Fred J. Damerau, eds.), CRC Press, Boca Raton, USA, 2nd ed., 2010.

[10] Iz Beltagy, Kyle Lo, and Arman Cohan, *SciBERT: A pretrained language model for scientific text*, Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP) (Hong Kong, China) (Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, eds.), Association for Computational Linguistics, November 2019, pp. 3615–3620.

[11] Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin, *A neural probabilistic language model*, Journal of machine learning research **3** (2003), no. Feb, 1137–1155.

[12] Giovanni Maria Biancofiore, Yashar Deldjoo, Tommaso Di Noia, Eugenio Di Sciascio, and Fedelucio Narducci, *Interactive question answering systems: Literature review*, ACM Computing Surveys **56** (2024), no. 9, 1–38.

[13] Joanne Boisson, Jose Camacho-Collados, and Luis Espinosa-Anke, *CardiffNLP-metaphor at SemEval-2022 task 2: Targeted fine-tuning of transformer-based language models for idiomaticity detection*, Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022) (Seattle, United States) (Guy Emerson, Natalie Schluter, Gabriel Stanovsky, Ritesh Kumar, Alexis Palmer, Nathan Schneider, Siddharth Singh, and Shyam Ratan, eds.), Association for Computational Linguistics, July 2022, pp. 169–177.

[14] Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov, *Enriching word vectors with subword information*, Transactions of the association for computational linguistics **5** (2017), 135–146.

[15] Dhouha Bouamor, Nasredine Semmar, and Pierre Zweigenbaum, *Identifying bilingual multi-word expressions for statistical machine translation*, Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12) (Istanbul, Turkey) (Nicoletta Calzolari, Khalid Choukri, Thierry Declerck, Mehmet Uğur Doğan, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, eds.), European Language Resources Association (ELRA), May 2012, pp. 674–679.

[16] Peter F Brown, Vincent J Della Pietra, Peter V Desouza, Jennifer C Lai, and Robert L Mercer, *Class-based n-gram models of natural language*, Computational linguistics **18** (1992), no. 4, 467–480.

[17] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al., *Language models are few-shot learners*, Advances in neural information processing systems **33** (2020), 1877–1901.

[18] Kamil Bujel, Helen Yannakoudakis, and Marek Rei, *Zero-shot sequence labeling for transformer-based sentence classifiers*, Proceedings of the 6th Workshop on Representation Learning for NLP (RepL4NLP-2021) (Online) (Anna Rogers, Iacer Calixto, Ivan Vulić, Naomi Saphra, Nora Kassner, Oana-Maria Camburu, Trapit Bansal, and Vered Shwartz, eds.), Association for Computational Linguistics, August 2021, pp. 195–205.

[19] Laura Castro, Anna Temerko, and Marcos Garcia, *Compositionality and ambiguity in multiword expressions: A dataset for the evaluation of language models in galician*, EPIA Conference on Artificial Intelligence, Springer, 2024, pp. 228–240.

[20] Paisarn Charoenpornsawat, Virach Sornlertlamvanich, and Thatsanee Charoenporn, *Improving translation quality of rule-based machine translation*, COLING-02: Machine Translation in Asia, 2002.

[21] Jiaao Chen and Diyi Yang, *Controllable conversation generation with conversation structures via diffusion models*, Findings of the Association for Computational Linguistics: ACL 2023 (Toronto, Canada) (Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, eds.), Association for Computational Linguistics, July 2023, pp. 7238–7251.

[22] Kyunghyun Cho, Bart van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio, *On the properties of neural machine translation: Encoder–decoder approaches*, Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation (Doha, Qatar) (Dekai Wu, Marine Carpuat, Xavier Carreras, and Eva Maria Vecchi, eds.), Association for Computational Linguistics, October 2014, pp. 103–111.

[23] Hyunjin Choi, Judong Kim, Seongho Joe, Seungjai Min, and Youngjune Gwon, *Analyzing zero-shot cross-lingual transfer in supervised nlp tasks*, 2020 25th International Conference on Pattern Recognition (ICPR), IEEE, 2021, pp. 9608–9613.

[24] Oswald Christopher et al., *Machine translation with large language models: Decoder only vs. encoder-decoder*, arXiv preprint arXiv:2409.13747 (2024).

[25] Zheng Chu, Ziqing Yang, Yiming Cui, Zhigang Chen, and Ming Liu, *HIT at SemEval-2022 task 2: Pre-trained language model for idioms detection*, Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022) (Seattle, United States) (Guy Emerson, Natalie Schluter, Gabriel Stanovsky, Ritesh Kumar, Alexis Palmer, Nathan Schneider, Siddharth Singh, and Shyam Ratan, eds.), Association for Computational Linguistics, July 2022, pp. 221–227.

[26] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio, *Empirical evaluation of gated recurrent neural networks on sequence modeling*, arXiv preprint arXiv:1412.3555 (2014).

[27] Jonathan H. Clark, Eunsol Choi, Michael Collins, Dan Garrette, Tom Kwiatkowski, Vitaly Nikolaev, and Jennimaria Palomaki, *TyDi QA: A benchmark for information-seeking question answering in typologically diverse lan-*

guages, Transactions of the Association for Computational Linguistics **8** (2020), 454–470.

[28] Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning, *Electra: Pre-training text encoders as discriminators rather than generators*, arXiv preprint arXiv:2003.10555 (2020).

[29] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov, *Unsupervised cross-lingual representation learning at scale*, Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (Online) (Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, eds.), Association for Computational Linguistics, July 2020, pp. 8440–8451.

[30] Alexis Conneau and Guillaume Lample, *Cross-lingual language model pretraining*, Advances in neural information processing systems **32** (2019).

[31] Mathieu Constant, Gülşen Eryiğit, Johanna Monti, Lonneke Van Der Plas, Carlos Ramisch, Michael Rosner, and Amalia Todirascu, *Multiword expression processing: A survey*, Computational Linguistics **43** (2017), no. 4, 837–892.

[32] Mathieu Constant, Gülşen Eryiğit, Johanna Monti, Lonneke van der Plas, Carlos Ramisch, Michael Rosner, and Amalia Todirascu, *Survey: Multiword expression processing: A Survey*, Computational Linguistics **43** (2017), no. 4, 837–892.

[33] Matthieu Constant, Anthony Sigogne, and Patrick Watrin, *Discriminative strategies to integrate multiword expression recognition and parsing*, Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (Jeju Island, Korea) (Haizhou Li, Chin-Yew

Lin, Miles Osborne, Gary Geunbae Lee, and Jong C. Park, eds.), Association for Computational Linguistics, July 2012, pp. 204–212.

[34] Silvio Cordeiro, Aline Villavicencio, Marco Idiart, and Carlos Ramisch, *Unsupervised compositionality prediction of nominal compounds*, Computational Linguistics **45** (2019), no. 1, 1–57.

[35] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova, *BERT: pre-training of deep bidirectional transformers for language understanding*, CoRR **abs/1810.04805** (2018).

[36] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al., *An image is worth 16x16 words: Transformers for image recognition at scale*, arXiv preprint arXiv:2010.11929 (2020).

[37] Marie Dubremetz and Joakim Nivre, *Extraction of nominal multiword expressions in French*, Proceedings of the 10th Workshop on Multiword Expressions (MWE) (Gothenburg, Sweden) (Valia Kordoni, Markus Egg, Agata Savary, Eric Wehrli, and Stefan Evert, eds.), Association for Computational Linguistics, April 2014, pp. 72–76.

[38] Stefan Evert, *The statistics of word cooccurrences: word pairs and collocations*, (2005).

[39] Samin Fakharian and Paul Cook, *Contextualized embeddings encode monolingual and cross-lingual knowledge of idiomaticity*, Proceedings of the 17th Workshop on Multiword Expressions (MWE 2021) (Online) (Paul Cook, Jelena Mitrović, Carla Parra Escartín, Ashwini Vaidya, Petya Osenova, Shiva Taslimipoor, and Carlos Ramisch, eds.), Association for Computational Linguistics, August 2021, pp. 23–32.

[40] Afsaneh Fazly, Paul Cook, and Suzanne Stevenson, *Unsupervised type and token identification of idiomatic expressions*, Computational Linguistics **35** (2009), no. 1, 61–103.

[41] Marcos Garcia, Tiago Kramer Vieira, Carolina Scarton, Marco Idiart, and Aline Villavicencio, *Assessing the representations of idiomaticity in vector models with a noun compound dataset labeled at type and token levels*, 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) (Online) (Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, eds.), Association for Computational Linguistics, August 2021, pp. 2730–2741.

[42] Nadia Mushtaq Gardazi, Ali Daud, Muhammad Kamran Malik, Amal Bukhari, Tariq Alsahfi, and Bader Alshemaimri, *Bert applications in natural language processing: a review*, Artificial Intelligence Review **58** (2025), no. 6, 1–49.

[43] Muskan Garg, Seema Wazarkar, Muskaan Singh, and Ondřej Bojar, *Multimodality for NLP-centered applications: Resources, advances and frontiers*, Proceedings of the Thirteenth Language Resources and Evaluation Conference (Marseille, France) (Nicoletta Calzolari, Frédéric Béchet, Philippe Blache, Khalid Choukri, Christopher Cieri, Thierry Declerck, Sara Goggi, Hitoshi Isahara, Bente Maegaard, Joseph Mariani, Hélène Mazo, Jan Odijk, and Stelios Piperidis, eds.), European Language Resources Association, June 2022, pp. 6837–6847.

[44] Eduardo C Garrido-Merchan, Roberto Gozalo-Brizuela, and Santiago Gonzalez-Carvajal, *Comparing bert against traditional machine learning models in text classification*, Journal of Computational and Cognitive Engineering **2** (2023), no. 4, 352–356.

[45] Waseem Gharbieh, Virendrakumar Bhavsar, and Paul Cook, *Deep learning models for multiword expression identification*, Proceedings of the 6th Joint Conference on Lexical and Computational Semantics (*SEM 2017) (Vancouver, Canada) (Nancy Ide, Aurélie Herbelot, and Lluís Màrquez, eds.), Association for Computational Linguistics, August 2017, pp. 54–64.

[46] Aaron Gluck, Katharina von der Wense, and Maria Leonor Pacheco, *Clix: Cross-lingual explanations of idiomatic expressions*, arXiv preprint arXiv:2501.03191 (2025).

[47] Yoav Goldberg, *Neural network methods in natural language processing*, Morgan & Claypool Publishers, 2017.

[48] Sebastian Gombert and Sabine Bartsch, *MultiVitaminBooster at PARSEME shared task 2020: Combining window- and dependency-based features with multilingual contextualised word embeddings for VMWE detection*, Proceedings of the Joint Workshop on Multiword Expressions and Electronic Lexicons (online) (Stella Markantonatou, John McCrae, Jelena Mitrović, Carole Tiberius, Carlos Ramisch, Ashwini Vaidya, Petya Osenova, and Agata Savary, eds.), Association for Computational Linguistics, December 2020, pp. 149–155.

[49] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al., *The llama 3 herd of models*, arXiv preprint arXiv:2407.21783 (2024).

[50] Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith, *Don't stop pretraining: Adapt language models to domains and tasks*, Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (Online) (Dan Jurafsky, Joyce Chai,

Natalie Schluter, and Joel Tetreault, eds.), Association for Computational Linguistics, July 2020, pp. 8342–8360.

[51] Lifeng Han, Gareth Jones, and Alan Smeaton, *MultiMWE: Building a multilingual multi-word expression (MWE) parallel corpora*, Proceedings of the Twelfth Language Resources and Evaluation Conference (Marseille, France) (Nicoletta Calzolari, Frédéric Béchet, Philippe Blache, Khalid Choukri, Christopher Cieri, Thierry Declerck, Sara Goggi, Hitoshi Isahara, Bente Maegaard, Joseph Mariani, Hélène Mazo, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, eds.), European Language Resources Association, May 2020, pp. 2970–2979 (eng).

[52] Zeyu Han, Chao Gao, Jinyang Liu, Jeff Zhang, and Sai Qian Zhang, *Parameter-efficient fine-tuning for large models: A comprehensive survey*, arXiv preprint arXiv:2403.14608 (2024).

[53] Tahmid Hasan, Abhik Bhattacharjee, Md. Saiful Islam, Kazi Mubasshir, Yuan-Fang Li, Yong-Bin Kang, M. Sohel Rahman, and Rifat Shahriyar, *XL-sum: Large-scale multilingual abstractive summarization for 44 languages*, Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021 (Online) (Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, eds.), Association for Computational Linguistics, August 2021, pp. 4693–4703.

[54] Reyhaneh Hashempour and Aline Villavicencio, *Leveraging contextual embeddings and idiom principle for detecting idiomaticity in potentially idiomatic expressions*, Proceedings of the Workshop on the Cognitive Aspects of the Lexicon (Online) (Michael Zock, Emmanuele Chersoni, Alessandro Lenci, and Enrico Santus, eds.), Association for Computational Linguistics, December 2020, pp. 72–80.

[55] Bradley Hauer, Seeratpal Jaura, Talgat Omarov, and Grzegorz Kondrak, *UAlberta at SemEval 2022 task 2: Leveraging glosses and translations for multilingual idiomaticity detection*, Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022) (Seattle, United States) (Guy Emerson, Natalie Schluter, Gabriel Stanovsky, Ritesh Kumar, Alexis Palmer, Nathan Schneider, Siddharth Singh, and Shyam Ratan, eds.), Association for Computational Linguistics, July 2022, pp. 145–150.

[56] Sen He, Wentong Liao, Hamed R Tavakoli, Michael Yang, Bodo Rosenhahn, and Nicolas Pugeault, *Image captioning through image transformer*, Proceedings of the Asian conference on computer vision, 2020.

[57] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean, *Distilling the knowledge in a neural network*, arXiv preprint arXiv:1503.02531 (2015).

[58] Sepp Hochreiter, *The vanishing gradient problem during learning recurrent neural nets and problem solutions*, International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems **6** (1998), no. 02, 107–116.

[59] Sepp Hochreiter and Jürgen Schmidhuber, *Long short-term memory*, Neural computation **9** (1997), no. 8, 1735–1780.

[60] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly, *Parameter-efficient transfer learning for nlp*, International conference on machine learning, PMLR, 2019, pp. 2790–2799.

[61] Jeremy Howard and Sebastian Ruder, *Universal language model fine-tuning for text classification*, Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (Melbourne,

Australia) (Iryna Gurevych and Yusuke Miyao, eds.), Association for Computational Linguistics, July 2018, pp. 328–339.

[62] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al., *Lora: Low-rank adaptation of large language models.*, ICLR **1** (2022), no. 2, 3.

[63] Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson, *Xtreme: A massively multilingual multi-task benchmark for evaluating cross-lingual generalisation*, International conference on machine learning, PMLR, 2020, pp. 4411–4421.

[64] Sami Itkonen, Jörg Tiedemann, and Mathias Creutz, *Helsinki-NLP at SemEval-2022 task 2: A feature-based approach to multilingual idiomaticity detection*, Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022) (Seattle, United States) (Guy Emerson, Natalie Schluter, Gabriel Stanovsky, Ritesh Kumar, Alexis Palmer, Nathan Schneider, Siddharth Singh, and Shyam Ratan, eds.), Association for Computational Linguistics, July 2022, pp. 122–134.

[65] Ray Jackendoff, *The architecture of the language faculty*, no. 28, mit Press, 1997.

[66] Guillaume Jacquet, Maud Ehrmann, Jakub Piskorski, Hristo Tanev, Ralf Steinberger, et al., *Cross-lingual linking of multi-word entities and language-dependent learning of multi-word entity patterns*, Representation and parsing of multiword expressions: Current trends **3** (2019), 269.

[67] Abhinav Jha, Hemprasad Yashwant Patil, Sumit Kumar Jindal, and Sardar MN Islam, *Multilingual indian language neural machine translation system using mt5 transformer*, 2023 2nd International Conference on Paradigm Shifts

in Communications Embedded Systems, Machine Learning and Signal Processing (PCEMS), IEEE, 2023, pp. 1–5.

[68] Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S Weld, Luke Zettlemoyer, and Omer Levy, *Spanbert: Improving pre-training by representing and predicting spans*, Transactions of the association for computational linguistics **8** (2020), 64–77.

[69] Daniel Jurafsky and James H. Martin, *Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition with language models*, 3rd ed., 2025, Online manuscript released January 12, 2025.

[70] Mihir Kale, Aditya Siddhant, Rami Al-Rfou, Linting Xue, Noah Constant, and Melvin Johnson, *nmT5 - is parallel data still relevant for pre-training massively multilingual language models?*, (2021), 683–691.

[71] Douwe Kiela and Stephen Clark, *Detecting compositionality of multi-word expressions using nearest neighbours in vector space models*, Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (Seattle, Washington, USA) (David Yarowsky, Timothy Baldwin, Anna Korhonen, Karen Livescu, and Steven Bethard, eds.), Association for Computational Linguistics, October 2013, pp. 1427–1432.

[72] Su Nam Kim, Timothy Baldwin, and Min-Yen Kan, *Extracting domain-specific words-a statistical approach*, Proceedings of the Australasian Language Technology Association Workshop 2009, 2009, pp. 94–98.

[73] Diederik P Kingma and Jimmy Ba, *Adam: A method for stochastic optimization*, arXiv preprint arXiv:1412.6980 (2014).

[74] Taku Kudo and John Richardson, *SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing*, Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations (Brussels, Belgium) (Eduardo Blanco and Wei Lu, eds.), Association for Computational Linguistics, November 2018, pp. 66–71.

[75] Sandeep Kumar and Arun Solanki, *An abstractive text summarization technique using transformer model with self-attention mechanism*, Neural Computing and Applications **35** (2023), no. 25, 18603–18622.

[76] Murathan Kurfalı, *TRAVIS at PARSEME shared task 2020: How good is (m)BERT at seeing the unseen?*, Proceedings of the Joint Workshop on Multiword Expressions and Electronic Lexicons (online) (Stella Markantonatou, John McCrae, Jelena Mitrović, Carole Tiberius, Carlos Ramisch, Ashwini Vaidya, Petya Osenova, and Agata Savary, eds.), Association for Computational Linguistics, December 2020, pp. 136–141.

[77] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut, *ALBERT: A lite BERT for self-supervised learning of language representations*, 8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020, 2020.

[78] Anne Lauscher, Vinit Ravishankar, Ivan Vulić, and Goran Glavaš, *From zero to hero: On the limitations of zero-shot language transfer with multilingual Transformers*, Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP) (Online) (Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, eds.), Association for Computational Linguistics, November 2020, pp. 4483–4499.

[79] Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang, *Biobert: a pre-trained biomedical language representation model for biomedical text mining*, Bioinformatics **36** (2020), no. 4, 1234–1240.

[80] Patrick Lewis, Barlas Oguz, Ruty Rinott, Sebastian Riedel, and Holger Schwenk, *MLQA: Evaluating cross-lingual extractive question answering*, Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (Online) (Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel Tetreault, eds.), Association for Computational Linguistics, July 2020, pp. 7315–7330.

[81] Junyi Li, Tianyi Tang, Wayne Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen, *Pre-trained language models for text generation: A survey*, ACM Computing Surveys **56** (2024), no. 9, 1–39.

[82] Xiang Lisa Li and Percy Liang, *Prefix-tuning: Optimizing continuous prompts for generation*, 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) (Online) (Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, eds.), Association for Computational Linguistics, August 2021, pp. 4582–4597.

[83] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig, *Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing*, ACM computing surveys **55** (2023), no. 9, 1–35.

[84] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov, *Roberta: A*

robustly optimized bert pretraining approach, arXiv preprint arXiv:1907.11692 (2019).

[85] Ilya Loshchilov and Frank Hutter, *Decoupled weight decay regularization*, arXiv preprint arXiv:1711.05101 (2017).

[86] Gyri Smørdal Losnegaard, Federico Sangati, Carla Parra Escartín, Agata Savary, Sascha Bargmann, and Johanna Monti, *PARSEME survey on MWE resources*, Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16) (Portorož, Slovenia) (Nicoletta Calzolari, Khalid Choukri, Thierry Declerck, Sara Goggi, Marko Grobelnik, Bente Maegaard, Joseph Mariani, Helene Mazo, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, eds.), European Language Resources Association (ELRA), May 2016, pp. 2299–2306.

[87] Daming Lu, *daminglu123 at SemEval-2022 task 2: Using BERT and LSTM to do text classification*, Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022) (Seattle, United States) (Guy Emerson, Natalie Schluter, Gabriel Stanovsky, Ritesh Kumar, Alexis Palmer, Nathan Schneider, Siddharth Singh, and Shyam Ratan, eds.), Association for Computational Linguistics, July 2022, pp. 186–189.

[88] Thang Luong, Hieu Pham, and Christopher D. Manning, *Effective approaches to attention-based neural machine translation*, Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (Lisbon, Portugal) (Lluís Màrquez, Chris Callison-Burch, and Jian Su, eds.), Association for Computational Linguistics, September 2015, pp. 1412–1421.

[89] Shuming Ma, Jian Yang, Haoyang Huang, Zewen Chi, Li Dong, Dongdong Zhang, Hany Hassan Awadalla, Alexandre Muzio, Akiko Eriguchi, Saksham

Singhal, et al., *Xlm-t: Scaling up multilingual machine translation with pretrained cross-lingual transformer encoders*, arXiv preprint arXiv:2012.15547 (2020).

[90] Christopher Manning and Hinrich Schutze, *Foundations of statistical natural language processing*, MIT press, 1999.

[91] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean, *Efficient estimation of word representations in vector space*, arXiv preprint arXiv:1301.3781 (2013).

[92] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean, *Distributed representations of words and phrases and their compositionality*, Advances in neural information processing systems **26** (2013).

[93] Shervin Minaee, Tomas Mikolov, Narjes Nikzad, Meysam Chenaghlu, Richard Socher, Xavier Amatriain, and Jianfeng Gao, *Large language models: A survey, 2024*, arXiv preprint arXiv:2402.06196 (2024).

[94] István Nagy T., Gábor Berend, and Veronika Vincze, *Noun compound and named entity recognition and their usability in keyphrase extraction*, Proceedings of the International Conference Recent Advances in Natural Language Processing 2011 (Hissar, Bulgaria) (Ruslan Mitkov and Galia Angelova, eds.), Association for Computational Linguistics, September 2011, pp. 162–169.

[95] Khalid Nassiri and Moulay Akhloufi, *Transformer models used for text-based question answering systems*, Applied Intelligence **53** (2023), no. 9, 10602–10635.

[96] Farhad Nooralahzadeh, Giannis Bekoulis, Johannes Bjerva, and Isabelle Augenstein, *Zero-shot cross-lingual transfer with meta learning*, Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing

(EMNLP) (Online) (Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, eds.), Association for Computational Linguistics, November 2020, pp. 4547–4562.

[97] Minsik Oh, *kpfriends at SemEval-2022 task 2: NEAMER - named entity augmented multi-word expression recognizer*, Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022) (Seattle, United States) (Guy Emerson, Natalie Schluter, Gabriel Stanovsky, Ritesh Kumar, Alexis Palmer, Nathan Schneider, Siddharth Singh, and Shyam Ratan, eds.), Association for Computational Linguistics, July 2022, pp. 178–185.

[98] Naoki Otani, Satoru Ozaki, Xingyuan Zhao, Yucen Li, Micaelah St Johns, and Lori Levin, *Pre-tokenization of multi-word expressions in cross-lingual word embeddings*, Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP) (Online) (Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, eds.), Association for Computational Linguistics, November 2020, pp. 4451–4464.

[99] Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, Lukasz Kaiser, Noam Shazeer, Alexander Ku, and Dustin Tran, *Image transformer*, International conference on machine learning, PMLR, 2018, pp. 4055–4064.

[100] Pavel Pecina, *A machine learning approach to multiword expression extraction*, Proceedings of the LREC Workshop Towards a Shared Task for Multiword Expressions (MWE 2008), vol. 2008, Marrakech,[s. p.], 2008, pp. 54–61.

[101] Jeffrey Pennington, Richard Socher, and Christopher Manning, *GloVe: Global vectors for word representation*, Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP) (Doha, Qatar) (Alessandro Moschitti, Bo Pang, and Walter Daelemans, eds.), Association for Computational Linguistics, October 2014, pp. 1532–1543.

[102] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer, *Deep contextualized word representations*, Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers) (New Orleans, Louisiana) (Marilyn Walker, Heng Ji, and Amanda Stent, eds.), Association for Computational Linguistics, June 2018, pp. 2227–2237.

[103] Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych, *AdapterFusion: Non-destructive task composition for transfer learning*, Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume (Online) (Paola Merlo, Jorg Tiedemann, and Reut Tsarfaty, eds.), Association for Computational Linguistics, April 2021, pp. 487–503.

[104] Jonas Pfeiffer, Ivan Vulić, Iryna Gurevych, and Sebastian Ruder, *MAD-X: An Adapter-Based Framework for Multi-Task Cross-Lingual Transfer*, Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP) (Online) (Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, eds.), Association for Computational Linguistics, November 2020, pp. 7654–7673.

[105] Thomas Pickard, Aline Villavicencio, Maggie Mi, Wei He, Dylan Phelps, Carolina Scarton, and Marco Idiart, *Semeval-2025 task 1: Admire–advancing multimodal idiomaticity representation*, arXiv preprint arXiv:2503.15358 (2025).

[106] Telmo Pires, Eva Schlinger, and Dan Garrette, *How multilingual is multilingual BERT?*, (2019), 4996–5001.

[107] Zhenting Qi, Hongyin Luo, Xuliang Huang, Zhuokai Zhao, Yibo Jiang, Xiangjun Fan, Himabindu Lakkaraju, and James R Glass, *Quantifying gener-*

alization complexity for large language models, The Thirteenth International Conference on Learning Representations, 2025.

[108] Lawrence R Rabiner, *A tutorial on hidden markov models and selected applications in speech recognition*, Proceedings of the IEEE **77** (1989), no. 2, 257–286.

[109] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al., *Improving language understanding by generative pre-training*, OpenAI blog (2018).

[110] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al., *Language models are unsupervised multitask learners*, OpenAI blog (2019).

[111] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu, *Exploring the limits of transfer learning with a unified text-to-text transformer*, Journal of machine learning research **21** (2020), no. 140, 1–67.

[112] Carlos Ramisch, *Multiword expressions acquisition*, A Generic and Open Framework. Cham: Springer International Publishing (2015).

[113] Carlos Ramisch, Silvio Cordeiro, Agata Savary, Veronika Vincze, Verginica Barbu Mititelu, Archna Bhatia, Maja Buljan, Marie Candito, Carla Parra Escartín, Polona Gantar, et al., *Edition 1.1 of the parseme shared task on automatic identification of verbal multiword expressions*, Proceedings of the joint workshop on linguistic annotation, multiword expressions and constructions (LAW-MWE-CxG-2018), USA, 2018, pp. 222–240.

[114] Carlos Ramisch, Silvio Ricardo Cordeiro, Agata Savary, Veronika Vincze, Verginica Barbu Mititelu, Archna Bhatia, Maja Buljan, Marie Candito, Polona Gantar, Voula Giouli, Tunga Güngör, Abdelati Hawwari, Uxoa Iñurrieta,

Jolanta Kovalevskaitė, Simon Krek, Timm Lichte, Chaya Liebeskind, Johanna Monti, Carla Parra Escartín, Behrang QasemiZadeh, Renata Ramisch, Nathan Schneider, Ivelina Stoyanova, Ashwini Vaidya, and Abigail Walsh, *Edition 1.1 of the PARSEME shared task on automatic identification of verbal multiword expressions*, Proceedings of the Joint Workshop on Linguistic Annotation, Multiword Expressions and Constructions (LAW-MWE-CxG-2018) (Santa Fe, New Mexico, USA) (Agata Savary, Carlos Ramisch, Jena D. Hwang, Nathan Schneider, Melanie Andresen, Sameer Pradhan, and Miriam R. L. Petruck, eds.), Association for Computational Linguistics, August 2018, pp. 222–240.

[115] Carlos Ramisch, Agata Savary, Bruno Guillaume, Jakub Waszczuk, Marie Candito, Ashwini Vaidya, Verginica Barbu Mititelu, Archna Bhatia, Uxoa Iñurrieta, Voula Giouli, Tunga Güngör, Menghan Jiang, Timm Lichte, Chaya Liebeskind, Johanna Monti, Renata Ramisch, Sara Stymne, Abigail Walsh, and Hongzhi Xu, *Edition 1.2 of the PARSEME shared task on semi-supervised identification of verbal multiword expressions*, Proceedings of the Joint Workshop on Multiword Expressions and Electronic Lexicons (online) (Stella Markantonatou, John McCrae, Jelena Mitrović, Carole Tiberius, Carlos Ramisch, Ashwini Vaidya, Petya Osenova, and Agata Savary, eds.), Association for Computational Linguistics, December 2020, pp. 107–118.

[116] Anton Razzhigaev, Mikhail Salnikov, Valentin Malykh, Pavel Braslavski, and Alexander Panchenko, *A system for answering simple questions in multiple languages*, Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations) (Toronto, Canada) (Danushka Bollegala, Ruihong Huang, and Alan Ritter, eds.), Association for Computational Linguistics, July 2023, pp. 524–537.

[117] Siva Reddy, Diana McCarthy, and Suresh Manandhar, *An empirical study on*

compositionality in compound nouns, Proceedings of 5th International Joint Conference on Natural Language Processing (Chiang Mai, Thailand) (Haifeng Wang and David Yarowsky, eds.), Asian Federation of Natural Language Processing, November 2011, pp. 210–218.

[118] Zhixiang Ren, Yajuan Lü, Jie Cao, Qun Liu, and Yun Huang, *Improving statistical machine translation using domain bilingual multiword expressions*, Proceedings of the Workshop on Multiword Expressions: Identification, Interpretation, Disambiguation and Applications (MWE 2009) (Singapore) (Dimitra Anastasiou, Chikara Hashimoto, Preslav Nakov, and Su Nam Kim, eds.), Association for Computational Linguistics, August 2009, pp. 47–54.

[119] Victoria Rosén, Koenraad De Smedt, Gyri Smørdal Losnegaard, Eduard Bejček, Agata Savary, and Petya Osenova, *MWEs in treebanks: From survey to guidelines*, Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16) (Portorož, Slovenia) (Nicoletta Calzolari, Khalid Choukri, Thierry Declerck, Sara Goggi, Marko Grobelnik, Bente Maegaard, Joseph Mariani, Helene Mazo, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, eds.), European Language Resources Association (ELRA), May 2016, pp. 2323–2330.

[120] Sebastian Ruder, Matthew E Peters, Swabha Swayamdipta, and Thomas Wolf, *Transfer learning in natural language processing*, Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: Tutorials, 2019, pp. 15–18.

[121] Sebastian Ruder, Jonas Pfeiffer, and Ivan Vulić, *Modular and parameter-efficient fine-tuning for nlp models*, Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing: Tutorial Abstracts, 2022, pp. 23–29.

[122] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams, *Learning representations by back-propagating errors*, Nature **323** (1986), no. 6088, 533–536.

[123] Ivan A Sag, Timothy Baldwin, Francis Bond, Ann Copestake, and Dan Flickinger, *Multiword expressions: A pain in the neck for nlp*, Computational Linguistics and Intelligent Text Processing: Third International Conference, CICLing 2002 Mexico City, Mexico, February 17–23, 2002 Proceedings 3, Springer, 2002, pp. 1–15.

[124] Marianne Santaholma, *Grammar sharing techniques for rule-based multilingual NLP systems*, Proceedings of the 16th Nordic Conference of Computational Linguistics (NODALIDA 2007) (Tartu, Estonia) (Joakim Nivre, Heiki-Jaan Kaalep, Kadri Muischnek, and Mare Koit, eds.), University of Tartu, Estonia, May 2007, pp. 253–260.

[125] Agata Savary, Cherifa Ben Khelil, Carlos Ramisch, Voula Giouli, Verginica Barbu Mititelu, Najet Hadj Mohamed, Cvetana Krstev, Chaya Liebeskind, Hongzhi Xu, Sara Stymne, Tunga Güngör, Thomas Pickard, Bruno Guillaume, Eduard Bejček, Archna Bhatia, Marie Candito, Polona Gantar, Uxoa Iñurrieta, Albert Gatt, Jolanta Kovalevskaite, Timm Lichte, Nikola Ljubešić, Johanna Monti, Carla Parra Escartín, Mehrnoush Shamsfard, Ivelina Stoyanova, Veronika Vincze, and Abigail Walsh, *PARSEME corpus release 1.3*, Proceedings of the 19th Workshop on Multiword Expressions (MWE 2023) (Dubrovnik, Croatia) (Archna Bhatia, Kilian Evang, Marcos Garcia, Voula Giouli, Lifeng Han, and Shiva Taslimipoor, eds.), Association for Computational Linguistics, May 2023, pp. 24–35.

[126] Agata Savary, Silvio Cordeiro, and Carlos Ramisch, *Without lexicons, multiword expression identification will never fly: A position statement*, Proceedings

of the Joint Workshop on Multiword Expressions and WordNet (MWE-WN 2019) (Florence, Italy) (Agata Savary, Carla Parra Escartín, Francis Bond, Jelena Mitrović, and Verginica Barbu Mititelu, eds.), Association for Computational Linguistics, August 2019, pp. 79–91.

[127] Agata Savary, Carlos Ramisch, Silvio Cordeiro, Federico Sangati, Veronika Vincze, Behrang QasemiZadeh, Marie Candito, Fabienne Cap, Voula Giouli, Ivelina Stoyanova, and Antoine Doucet, *The PARSEME shared task on automatic identification of verbal multiword expressions*, Proceedings of the 13th Workshop on Multiword Expressions (MWE 2017) (Valencia, Spain) (Stella Markantonatou, Carlos Ramisch, Agata Savary, and Veronika Vincze, eds.), Association for Computational Linguistics, April 2017, pp. 31–47.

[128] Dominik Schlechtweg, Anna Hätty, Marco Del Tredici, and Sabine Schulte im Walde, *A wind of change: Detecting and evaluating lexical semantic change across times and domains*, Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (Florence, Italy) (Anna Korhonen, David Traum, and Lluís Màrquez, eds.), Association for Computational Linguistics, July 2019, pp. 732–746.

[129] Rico Sennrich, Barry Haddow, and Alexandra Birch, *Improving neural machine translation models with monolingual data*, Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (Berlin, Germany) (Katrin Erk and Noah A. Smith, eds.), Association for Computational Linguistics, August 2016, pp. 86–96.

[130] ———, *Neural machine translation of rare words with subword units*, Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (Berlin, Germany) (Katrin Erk and

Noah A. Smith, eds.), Association for Computational Linguistics, August 2016, pp. 1715–1725.

[131] Yutaro Shigeto, Ai Azuma, Sorami Hisamoto, Shuhei Kondo, Tomoya Kose, Keisuke Sakaguchi, Akifumi Yoshimoto, Frances Yung, and Yuji Matsumoto, *Construction of English MWE dictionary and its application to POS tagging*, Proceedings of the 9th Workshop on Multiword Expressions (Atlanta, Georgia, USA) (Valia Kordoni, Carlos Ramisch, and Aline Villavicencio, eds.), Association for Computational Linguistics, June 2013, pp. 139–144.

[132] Xinying Song, Alex Salcianu, Yang Song, Dave Dopson, and Denny Zhou, *Fast WordPiece tokenization*, Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (Online and Punta Cana, Dominican Republic) (Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, eds.), Association for Computational Linguistics, November 2021, pp. 2089–2103.

[133] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov, *Dropout: a simple way to prevent neural networks from overfitting*, The journal of machine learning research **15** (2014), no. 1, 1929–1958.

[134] Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang, *How to fine-tune bert for text classification?*, China national conference on Chinese computational linguistics, Springer, 2019, pp. 194–206.

[135] Jimin Sun, Hwijeen Ahn, Chan Young Park, Yulia Tsvetkov, and David R. Mortensen, *Cross-cultural similarity features for cross-lingual transfer learning of pragmatically motivated tasks*, Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main

Volume (Online) (Paola Merlo, Jorg Tiedemann, and Reut Tsarfaty, eds.), Association for Computational Linguistics, April 2021, pp. 2403–2414.

[136] Lin Sun, Jiquan Wang, Kai Zhang, Yindu Su, and Fangsheng Weng, *Rpbert: a text-image relation propagation-based bert model for multimodal ner*, Proceedings of the AAAI conference on artificial intelligence, vol. 35, 2021, pp. 13860–13868.

[137] Raghuraman Swaminathan and Paul Cook, *Token-level identification of multiword expressions using pre-trained multilingual language models*, Proceedings of the 19th Workshop on Multiword Expressions (MWE 2023) (Dubrovnik, Croatia) (Archna Bhatia, Kilian Evang, Marcos Garcia, Voula Giouli, Lifeng Han, and Shiva Taslimipoor, eds.), Association for Computational Linguistics, May 2023, pp. 1–6.

[138] Nima Tajbakhsh, Jae Y Shin, Suryakanth R Gurudu, R Todd Hurst, Christopher B Kendall, Michael B Gotway, and Jianming Liang, *Convolutional neural networks for medical image analysis: Full training or fine tuning?*, IEEE transactions on medical imaging **35** (2016), no. 5, 1299–1312.

[139] Minghuan Tan, *HiJoNLP at SemEval-2022 task 2: Detecting idiomaticity of multiword expressions using multilingual pretrained language models*, Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022) (Seattle, United States) (Guy Emerson, Natalie Schluter, Gabriel Stanovsky, Ritesh Kumar, Alexis Palmer, Nathan Schneider, Siddharth Singh, and Shyam Ratan, eds.), Association for Computational Linguistics, July 2022, pp. 190–196.

[140] Shiva Taslimipoor, Sara Bahaadini, and Ekaterina Kochmar, *MTLB-STRUCT @parseme 2020: Capturing unseen multiword expressions using multi-task*

learning and pre-trained masked language models, Proceedings of the Joint Workshop on Multiword Expressions and Electronic Lexicons (online) (Stella Markantonatou, John McCrae, Jelena Mitrović, Carole Tiberius, Carlos Ramisch, Ashwini Vaidya, Petya Osenova, and Agata Savary, eds.), Association for Computational Linguistics, December 2020, pp. 142–148.

[141] Shiva Taslimipoor, Omid Rohanian, and Le An Ha, *Cross-lingual transfer learning and multitask learning for capturing multiword expressions*, Proceedings of the Joint Workshop on Multiword Expressions and WordNet (MWE-WN 2019) (Florence, Italy) (Agata Savary, Carla Parra Escartín, Francis Bond, Jelena Mitrović, and Verginica Barbu Mititelu, eds.), Association for Computational Linguistics, August 2019, pp. 155–161.

[142] Harish Tayyar Madabushi, Edward Gow-Smith, Marcos Garcia, Carolina Scarton, Marco Idiart, and Aline Villavicencio, *SemEval-2022 task 2: Multilingual idiomaticity detection and sentence embedding*, Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022) (Seattle, United States) (Guy Emerson, Natalie Schluter, Gabriel Stanovsky, Ritesh Kumar, Alexis Palmer, Nathan Schneider, Siddharth Singh, and Shyam Ratan, eds.), Association for Computational Linguistics, July 2022, pp. 107–121.

[143] Simone Tedeschi and Roberto Navigli, *NER4ID at SemEval-2022 task 2: Named entity recognition for idiomaticity detection*, Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022) (Seattle, United States) (Guy Emerson, Natalie Schluter, Gabriel Stanovsky, Ritesh Kumar, Alexis Palmer, Nathan Schneider, Siddharth Singh, and Shyam Ratan, eds.), Association for Computational Linguistics, July 2022, pp. 204–210.

[144] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin, *Attention is all you need*,

Advances in neural information processing systems **30** (2017).

[145] Abigail Walsh, Claire Bonial, Kristina Geeraert, John P. McCrae, Nathan Schneider, and Clarissa Somers, *Constructing an annotated corpus of verbal MWEs for English*, Proceedings of the Joint Workshop on Linguistic Annotation, Multiword Expressions and Constructions (LAW-MWE-CxG-2018) (Santa Fe, New Mexico, USA) (Agata Savary, Carlos Ramisch, Jena D. Hwang, Nathan Schneider, Melanie Andresen, Sameer Pradhan, and Miriam R. L. Petruck, eds.), Association for Computational Linguistics, August 2018, pp. 193–200.

[146] Abigail Walsh, Teresa Lynn, and Jennifer Foster, *A BERT's eye view: Identification of Irish multiword expressions using pre-trained language models*, Proceedings of the 18th Workshop on Multiword Expressions @LREC2022 (Marseille, France) (Archna Bhatia, Paul Cook, Shiva Taslimipoor, Marcos Garcia, and Carlos Ramisch, eds.), European Language Resources Association, June 2022, pp. 89–99.

[147] Shuohang Wang, Luowei Zhou, Zhe Gan, Yen-Chun Chen, Yuwei Fang, Siqi Sun, Yu Cheng, and Jingjing Liu, *Cluster-former: Clustering-based sparse transformer for question answering*, Findings of the association for computational linguistics: ACL-IJCNLP 2021, 2021, pp. 3958–3968.

[148] Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush, *Transformers: State-of-the-art natural language processing*, (2020), 38–45.

[149] Alison Wray, *Formulaic language and the lexicon.*, ERIC, 2002.

[150] Shijie Wu and Mark Dredze, *Beto, bentz, becas: The surprising cross-lingual effectiveness of BERT*, Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP) (Hong Kong, China) (Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, eds.), Association for Computational Linguistics, November 2019, pp. 833–844.

[151] Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel, *mT5: A massively multilingual pre-trained text-to-text transformer*, Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Online) (Kristina Toutanova, Anna Rumshisky, Luke Zettlemoyer, Dilek Hakkani-Tur, Iz Beltagy, Steven Bethard, Ryan Cotterell, Tanmoy Chakraborty, and Yichao Zhou, eds.), Association for Computational Linguistics, June 2021, pp. 483–498.

[152] Atsuki Yamaguchi, Gaku Morio, Hiroaki Ozaki, and Yasuhiro Sogawa, *Hitachi at SemEval-2022 task 2: On the effectiveness of span-based classification approaches for multilingual idiomaticity detection*, Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022) (Seattle, United States) (Guy Emerson, Natalie Schluter, Gabriel Stanovsky, Ritesh Kumar, Alexis Palmer, Nathan Schneider, Siddharth Singh, and Shyam Ratan, eds.), Association for Computational Linguistics, July 2022, pp. 135–144.

[153] Yeqing Yan, Peng Zheng, and Yongjun Wang, *Enhancing large language model capabilities for rumor detection with knowledge-powered prompting*, Engineering Applications of Artificial Intelligence **133** (2024), 108259.

[154] Wei Yang, Yuqing Xie, Aileen Lin, Xingyu Li, Luchen Tan, Kun Xiong, Ming Li, and Jimmy Lin, *End-to-end open-domain question answering with BERT-*

serini, Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations) (Minneapolis, Minnesota) (Waleed Ammar, Annie Louis, and Nasrin Mostafazadeh, eds.), Association for Computational Linguistics, June 2019, pp. 72–77.

[155] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le, *Xlnet: Generalized autoregressive pretraining for language understanding*, Advances in neural information processing systems **32** (2019).

[156] Zeynep Yirmibeşoğlu and Tunga Güngör, *ERMI at PARSEME shared task 2020: Embedding-rich multiword expression identification*, Proceedings of the Joint Workshop on Multiword Expressions and Electronic Lexicons (online) (Stella Markantonatou, John McCrae, Jelena Mitrović, Carole Tiberius, Carlos Ramisch, Ashwini Vaidya, Petya Osenova, and Agata Savary, eds.), Association for Computational Linguistics, December 2020, pp. 130–135.

[157] Yue Yu, Simiao Zuo, Haoming Jiang, Wendi Ren, Tuo Zhao, and Chao Zhang, *Fine-tuning pre-trained language model with weak supervision: A contrastive-regularized self-training approach*, Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Online) (Kristina Toutanova, Anna Rumshisky, Luke Zettlemoyer, Dilek Hakkani-Tur, Iz Beltagy, Steven Bethard, Ryan Cotterell, Tanmoy Chakraborty, and Yichao Zhou, eds.), Association for Computational Linguistics, June 2021, pp. 1063–1077.

[158] Andrea Zaninello and Alexandra Birch, *Multiword expression aware neural machine translation*, Proceedings of the Twelfth Language Resources and Evaluation Conference (Marseille, France) (Nicoletta Calzolari, Frédéric Béchet, Philippe Blache, Khalid Choukri, Christopher Cieri, Thierry Declerck, Sara Goggi, Hitoshi Isahara, Bente Maegaard, Joseph Mariani, Hélène Mazo, Asun-

cion Moreno, Jan Odijk, and Stelios Piperidis, eds.), European Language Resources Association, May 2020, pp. 3816–3825 (eng).

[159] Hanqing Zhang, Haolin Song, Shaoyu Li, Ming Zhou, and Dawei Song, *A survey of controllable text generation using transformer-based pre-trained language models*, ACM Computing Surveys **56** (2023), no. 3, 1–37.

[160] Jiong Zhang, Wei-Cheng Chang, Hsiang-Fu Yu, and Inderjit Dhillon, *Fast multi-resolution transformer fine-tuning for extreme multi-label text classification*, Advances in Neural Information Processing Systems **34** (2021), 7267–7280.

[161] Linrui Zhang and Dan Moldovan, *Rule-based vs. neural net approaches to semantic textual similarity*, Proceedings of the First Workshop on Linguistic Resources for Natural Language Processing (Santa Fe, New Mexico, USA) (Peter Machonis, Anabela Barreiro, Kristina Kocijan, and Max Silberztein, eds.), Association for Computational Linguistics, August 2018, pp. 12–17.

[162] Shengyu Zhang, Linfeng Dong, Xiaoya Li, Sen Zhang, Xiaofei Sun, Shuhe Wang, Jiwei Li, Runyi Hu, Tianwei Zhang, Fei Wu, et al., *Instruction tuning for large language models: A survey*, arXiv preprint arXiv:2308.10792 (2023).

[163] Chen Zheng, Ke Sun, Hang Wu, Chenguang Xi, and Xun Zhou, *Balancing enhancement, harmlessness, and general capabilities: Enhancing conversational llms with direct rlhf*, arXiv preprint arXiv:2403.02513 (2024).

[164] Shuai Zheng, Sadeep Jayasumana, Bernardino Romera-Paredes, Vibhav Vineet, Zhizhong Su, Dalong Du, Chang Huang, and Philip HS Torr, *Conditional random fields as recurrent neural networks*, Proceedings of the IEEE international conference on computer vision, 2015, pp. 1529–1537.

[165] Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler, *Aligning books and movies: Towards story-like visual explanations by watching movies and reading books*, The IEEE International Conference on Computer Vision (ICCV), December 2015.

Vita

Candidate's full name: Md. Arid Hasan

University attended:

- Master of Computer Science, University of New Brunswick, Fredericton, NB, Canada (2023-2025)
- Bachelor of Science in Computer Science and Engineering, Daffodil International University, Dhaka, Bangladesh (2015-2019)

Publications:

- Basel Mousi, Nadir Durrani, Fatema Ahmad, **Md. Arid Hasan**, Maram Hasanain, Tameem Kabbani, Fahim Dalvi, Shammur Absar Chowdhury, and Firoj Alam. 2025. AraDiCE: Benchmarks for Dialectal and Cultural Capabilities in LLMs. In Proceedings of the 31st International Conference on Computational Linguistics, pages 4186–4218, Abu Dhabi, UAE. Association for Computational Linguistics.
- Maram Hasanain, **Md. Arid Hasan**, Fatema Ahmad, Reem Suwaileh, Md. Rafiul Biswas, Wajdi Zaghouani, and Firoj Alam. 2024. ArAIEval Shared Task: Propagandistic Techniques Detection in Unimodal and Multimodal Arabic Content. In Proceedings of the Second Arabic Natural Language Processing

Conference, pages 456–466, Bangkok, Thailand. Association for Computational Linguistics.

- Firoj Alam, Abul Hasnat, Fatema Ahmad, **Md. Arid Hasan**, and Maram Hasanain. 2024. ArMeme: Propagandistic Content in Arabic Memes. In Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, pages 21071–21090, Miami, Florida, USA. Association for Computational Linguistics.

- **Md. Arid Hasan**, Firoj Alam, Anika Anjum, Shudipta Das, and Afiyat Anjum. 2023. BLP-2023 Task 2: Sentiment Analysis. In Proceedings of the First Workshop on Bangla Language Processing (BLP-2023), pages 354–364, Singapore. Association for Computational Linguistics.

- **Md. Arid Hasan**, Shudipta Das, Afiyat Anjum, Firoj Alam, Anika Anjum, Avijit Sarker, and Sheak Rashed Haider Noori. 2024. Zero- and Few-Shot Prompting with LLMs: A Comparative Study with Fine-tuned Models for Bangla Sentiment Analysis. In Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024), pages 17808–17818, Torino, Italia. ELRA and ICCL.

- Firoj Alam, Kashif Ahmad, **Md. Arid Hasan**, Ferda Ofli, Muhammad Imran. (2023). Role of Social Media Imagery in Disaster Informatics. In: Singh, A. (eds) International Handbook of Disaster Research. Springer, Singapore.

- Firoj Alam, Tanvirul Alam, **Md. Arid Hasan**, et al. MEDIC: a multi-task learning dataset for disaster image classification. Neural Computing & Applications 35, 2609–2632 (2023).

- Roberto Zamparelli, Shammur Chowdhury, Dominique Brunato, Cristiano Chesi, Felice Dell'Orletta, **Md. Arid Hasan**, and Giulia Venturi. 2022.

SemEval-2022 Task 3: PreTENS-Evaluating Neural Networks on Presuppositional Semantic Knowledge. In Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022), pages 228–238, Seattle, United States. Association for Computational Linguistics.

- **Md. Arid Hasan**, Jannatul Tajrin, Shammur Absar Chowdhury, and Firoj Alam. Sentiment Classification in Bangla Textual Content: A Comparative Study. 2020 23rd International Conference on Computer and Information Technology (ICCIT), Dhaka, Bangladesh, 2020, pp. 1-6.

- **Md. Arid Hasan**, Firoj Alam, Shammur Absar Chowdhury, and Naira Khan. Neural Machine Translation for the Bangla-English Language Pair. 2019 22nd International Conference on Computer and Information Technology (ICCIT), Dhaka, Bangladesh, 2019, pp. 1-5.

- **Md. Arid Hasan**, Firoj Alam, Shammur Absar Chowdhury, and Naira Khan. Neural vs Statistical Machine Translation: Revisiting the Bangla-English Language Pair. 2019 International Conference on Bangla Speech and Language Processing (ICBSLP), Sylhet, Bangladesh, 2019, pp. 1-5.

- **Md. Arid Hasan**, Firoj Alam, Sheak Rashed Haider Noori. (2020). A Collaborative Platform to Collect Data for Developing Machine Translation Systems. In: Uddin, M.S., Bansal, J.C. (eds) Proceedings of International Joint Conference on Computational Intelligence. Algorithms for Intelligent Systems. Springer, Singapore.

Conference Presentations: N/A