LMU

LUDWIG-
MAXIMILIANS-
UNIVERSITÄT
MÜNCHEN

CENTRUM FÜR INFORMATIONS- UND SPRACHVERARBEITUNG
STUDIENGANG COMPUTERLINGUISTIK

Masterarbeit

im Studiengang Computerlinguistik

an der Ludwig- Maximilians- Universität München

Fakultät für Sprach- und Literaturwissenschaften

Zero-Shot Learning on Low-Resource Languages by Cross-Lingual Retrieval

vorgelegt von
Ercong Nie

Betreuer:	Sheng Liang
Prüfer:	Prof. Dr. Hinrich Schütze
Bearbeitungszeitraum:	21. März - 07. August 2022

Selbstständigkeitserklärung

Ich erkläre hiermit, dass ich die vorliegende Arbeit selbständig angefertigt, alle Zitate als solche kenntlich gemacht sowie alle benutzten Quellen und Hilfsmittel angegeben habe.

München, den 08. August 2022

....
Ercong Nie

Abstract

Research on zero-shot learning on low-resource languages in NLP is motivated by inherent data scarcity of low-resource languages. On the contrary, languages with a large amount of both labeled and unlabeled resources are not fully utilized. Multilingual Pretrained Language Models (MPLMs) have shown its strong multilinguality in recent empirical cross-lingual transfer studies. This research aims to improve the zero-shot transfer learning performance on low-resource languages by exploiting the rich annotated data and raw data of high-resource languages. In our work, we have applied our proposed approach to three tasks (binary sentiment classification, topic categorization, natural language inference) with multilingual parallel test sets across 10 low-resource languages covering 6 language families. We have also presented an empirical analysis on the effect of the number of cross-lingual retrieved samples and language properties. Our proposed method obtain a performance improvement of 33.6%, 118.6% and 8.7% in the setting of retrieval from labeled high-resource language corpora and 6.1%, 23.2% and 1.3% in the setting of retrieval from unlabeled high-resource language corpora on binary sentiment classification, topic categorization and XNLI tasks, respectively. Our study presents the potential to improve zero-shot transfer performance on low-resource languages by cross-lingual retrieval from high-resource languages.

Zusammenfassung

Die Forschung zum Zero-Shot-Lernen in ressourcenarmen Sprachen im NLP ist motiviert durch die inhärente Datenknappheit von ressourcenarmen Sprachen. Im Gegensatz dazu werden Sprachen mit einer großen Menge sowohl annotierter als auch unannotierter Ressourcen nicht vollständig genutzt. Multilinguale vortrainierte Sprachmodelle (MPLMs) haben ihre starke Mehrsprachigkeit in neueren empirischen spracheübergreifenden Transferstudien gezeigt. Diese Studie zielt auf die Zero-Shot-Transfer-Lernleistung bei ressourcenarmen Sprachen ab, indem die reichen annotierten Daten und Rohdaten von ressourcenreichen Sprachen genutzt werden. In unserer Arbeit haben wir unseren vorgeschlagenen Ansatz auf drei Aufgaben (binäre Sentimentklassifizierung, Themenkategorisierung, natürliche Sprachinferenz) mit multilingualen parallelen Testdatensätzen in 10 ressourcenarmen Sprachen angewendet, die 6 Sprachfamilien abdecken. Wir haben auch eine empirische Analyse zur Auswirkung der Anzahl der sprachübergreifend abgerufenen Beispiele und Spracheigenschaften vorgelegt. Unsere vorgeschlagene Methode erzielt eine Leistungsverbesserung von 33,6%, 118,6% und 8,7% in der Einstellung des Abrufs aus annotierten Korpora von ressourcenreichen Sprachen und 6,1%, 23,2% und 1,3% in der Einstellung des Abrufs aus unannotierten Korpora von ressourcenreichen Sprachen zur binären Sentimentklassifizierung, Themenkategorisierung bzw. XNLI-Aufgaben. Unsere Studie zeigt das Potenzial zur Verbesserung der Zero-Shot-Übertragungsleistung bei ressourcenarmen Sprachen durch sprachübergreifenden Abruf aus ressourcenreichen Sprachen.

Acknowledge

At this moment, at the very end of my Master study, I feel more than excited to write these words, because after getting over so much suffering and struggle in the difficult time of more than two years, I finally approach the peak of my current stage of life. Thinking back to two years ago, I was going to start my (online) Master study in anxiety, uncertainty as well as in pandemics at my home in China. As a Bachelor graduate with language background, I at that time chose an uncommon path to studying computational linguistics out of my naive interest in combining languages and technologies. I had neither solid mathematical fundamentals nor programming experience and was doubting on myself with fear whether I would manage to finish my Master study. At the end of my writing the Master thesis, I feel proud of myself that I make it! I gain more than I had expected. So I would like to pass the first thank to myself, to me who was not defeated by doubt and fear and bravely made the decision two years ago, to me who through his efforts turned his interest into professions and found his way to a higher level in the past two years. Trust yourself and keep moving on!

I would like to thank my examiner Prof. Hinrich Schütze and my supervisor Sheng Liang for their support and guidance through my work on the Master thesis. I would also like to thank all teachers at CIS who had taught me during the master study. The highly intensive practice courses from Dr. Helmut Schmid helped me a lot in the improvement of my coding ability. The seminars by Dr. Stefan Langer and Dr. Christoph Ringlstetter broadened my horizons in machine learning algorithms in NLP and the interesting NLP-relevant field of conversational AI. The informative machine translation course by Prof. Alexander Fraser was the first systematic NLP course that I have taken. The course by Dr. Robert Zangenfeind reminded me that I was a computational linguist rather than machine learning student. Thank Prof. Klaus U. Schulz for the trust on me to undertake the teaching work of a programming course.

I want to thank my great CIS fellow students. I appreciate the cooperation and discussions in many courses with Han-Ching, Haotian and Chunlan. A big thank to Haotian, who taught me so much useful and necessary technical stuff that I never knew before. Thank Shuzhou for giving me much useful guidance as a senior student.

The last but the most thank go to my parents, who have always unconditionally supported me for more than 25 years. I wish I would not let you down.

Table of Contents

1 Introduction

The rapid popularization of deep neural network methods in recent years has caused a revolutionary transformation from traditional fully supervised machine learning methods to the age of pretrained language models (PLMs). One remarkable symbol is the emergence of transformer architecture (Vaswani et al., 2017), which is a deep neural network model making full use of the self-attention mechanism. The emergence of transformer achitecture brings a novel paradigm to the natural language processing (NLP) research, which is the so-called pretraining-finetuning paradigm. This new paradigm has achieved inspiringly good performance in almost all NLP tasks, for example text classification, natural language inference, questions answering, machine translation etc. In the pretraining-finetuning paradigm, the transformer-based language model is firstly pretrained on large text corpora with different self-supervised pretraining objectives. For this step, merely raw language data is enough for training and annotated data is not needed. Then, the domain- or task-specific annotated data is used to finetune the parameters of the model so that the pretraiend model acquires the domain adaptability and can be used for a specific task. This pretraining-finetuning paradigm has gradually been a default to apply large-scale language models to concrete NLP tasks. However, most of the progress is based on a premise that massive language data could be collected used for pretraining, while this is not a good solution for all languages. According to Ethnologue (2021)[1], there are 7,139 living languages in the world nowadays, but only 1% of them are spoken by more than 10 million people. Therefore, most of the languages have very poor digital text resources. In fact, most of the state-of-the-art models are available in only resource-rich languages, such as English and Chinese. For those low-resource languages whose training data is rather scarce, it is still difficult to benefit from the PLMs.

Even for high-resource languages, the annotation work to produce sufficient data used for finetuning still requires a large amount of human resource. Besides, the annotation work has to be done for each individual task, and thus the generalization is lacked. With the development of even larger scale PLMs, it is increasingly assumed that the large PLMs have already learned enough language representation during the pretraining to solve downstream NLP tasks. We can activate its ability by reformulating the input so that a PLM can directly produce a proper output based on what it has learned from the pretraining. The idea of reformulating the input and directly utilizing PLM's language ability learned from pretraining is called prompt-based learning. The prompt is designed to help the PLM "understand" the task and recall what it has learned in the pretraining. In other words, by reformulating the input, we make the form of one downstream task look more similar to the pretraining tasks the PLM has done. Prompt-based learning offers a new form of zero-shot or few-shot learning and potentially serves as a proper method for low-resource languages lacking in annotated data.

The prevalence of multilingual pretrained language models (MPLMs) provides the low-resource languages a possibility to be profited by large-scale language models. Similar to PLMs, MPLMs are also based on transformer architectures and pretrained on large amount of language data. The difference lies that MPLMs are jointly pretrained on multilingual corpora. The sort of languages can achieve more than one hundred, such as multilingual BERT (mBERT) (Devlin et al., 2018), XLM (Conneau et al., 2019) etc. Multilingual contextualized language models have been proved to exhibit a great degree of multilinguality as measured for example by zero-shot cross-lingual transfer (Hu et al., 2020). However,

[1] https://www.ethnologue.com/guides/how-many-languages

not all languages are created equal in MPLMs (Wu and Dredze, 2020). Even in MPLMs, low-resource languages are still underpresented compared to the high-resource. Zero-shot cross-lingual transfer learning is a popular method of tackling NLP tasks on low-resource languages by utilizing MPLMs. Zero-shot cross-lingual transfer learning uses task-specific annotations of one language to fine-tune the model and then the finetuned model is directly applied to the same task of another language. However, low-resource languages usually perform weakly on multilingual benchmarks under the zero-shot cross-lingual setting (Lauscher et al., 2020).

1.1 Research Objective

The volume of text data has constrained low-resource languages to benefit more profoundly from the innovation wave of NLP technologies. On the one hand, the scarcity of domain- or task-specific annotated data makes the pretraining-finetuning paradigm almost impossible for low-resource languages. On the other hand, the deficiency of raw language data of low-resource languages leads to an imbalanced distribution of language data in the MPLMs' multilingual pretraining corpora. This causes that the low-resource languages are underrepresented by the MPLMs during the pretraining step so that bad performance is achieved on low-resource languages by zero-shot cross-lingual transfer learning methods.

Based on above, this work aims to improve the performance of low-resource languages on NLP tasks by taking advantage of the multilinguality of MPLMs and cross-lingual information retrieval. Motivated by the prompt-based learning, we propose a multilingual prompt engineering specifically for the MPLMs. In our method, the prompt is designed by adding retrieved cross-lingual information. By applying cross-lingual prompt to the MPLM, we try to activate the multilinguality and cross-lingual transfer ability of MPLM.

This work focuses on combining several different types of advanced NLP methods and tries to find a proper solution for low-resource languages in the setting of zero-shot learning. To validate the idea, we design and conduct experiments on several types of text classification tasks with different approaches of cross-lingual retrieval as well as prompt engineering. We expect that the low-resource languages would benefit from this method and thereby achieve better performance.

1.2 Research Questions

- How to utilize retrieved cross-lingual information from both labeled and unlabeled high-resource languages such that the zero-shot transfer performance on low-resource languages can be significantly improved?
- How to conduct cross-lingual zero-shot transfer evaluation on as many low-resource languages as possible in spite of the scarcity of low-resource language resources?
- Which linguistic factors could affect the zero-shot transfer performance? How is the correlation between these factors and performance like?
- How will the amount of cross-lingual retrieved information influence the zero-shot performance on low-resource languages?

1.3 Contributions of the Work

- We have proposed an approach of integrating retrieved cross-lingual information into prompting engineering for zero-shot transfer learning. We have proved with experiments on different tasks that zero-shot transfer learning performance on low-resource languages can be improved by cross-lingual retrieval from both labeled and unlabeled high-resource language corpora. The improvement with labeled corpora is better than that with unlabeled corpora (+33.6% vs. +6.1% on Amazon review and +118.6% vs. +23.2%).

- We have expanded the test data of different tasks to more low-resource languages by creating a multilingual parallel datasets using machine translation.
- We have presented that language similarity and pretraining corpus size correlate to the zero-shot transfer performance with our method.
- We have proved that The number of cross-lingual retrieved samples affect the performance. Adding best matched cross-lingual sample as priming information brings drastic rise in accuracy. However, continuing to increase k will slow down the performance increasing. The performance will gradually flatten and even shrink.

1.4 Organization of the Thesis

- Chapter 1 provides an introduction to the work. Research objective, research questions, research contributions are summarized in this chapter. The outline of the thesis is presented.
- In Chapter 2, we provide an overview of theoretical background that is relevant with the research topic in order to better understand the contents of this thesis. We review the development of pretrained language models (PLMs) and multilingual pretrained language models (MPLMs). We furthermore discuss the topic of language resources. We also cover the fundamentals of information retrieval.
- In Chapter 3, we review the works related to the research field in recent years. We profile the analysis of multilinguality and recent empirical studies on cross-lingual transfer learning. Besides, we explain the emerging prompt-based learning and sentence transformers in NLP field.
- Chapter 4 introduces our research methods. We first propose two pipelines for cross-lingual retrieval from labeled and unlabeled high-resource language data. In the following parts, we describe each module in both pipelines in detail.
- Chapter 5 presents the experimental settings. We introduce the datasets, languages and models that are used in our experiment in detail.
- In Chapter 6, we focus on the analysis of the experimental results. We provide an overview of the results on each task and then further detect the effect and correlations of relevant factors.
- Chapter 7 finally contains our conclusions where we summarize our findings and provide an outlook into the future work.

2 Theoretical Background

This chapter introduces the theoretical background of several different fields that are closely relevant with the research in this paper. Since the research objects of this work are multilingual language models, it is necessary to investigate the development of pretrained language models (PLMs) (2.1) as well as multilingual pretrained language models (MPLMs) (2.2) that are derived from PLMs.

This work focuses on the performance improvement of low-resource languages. In Section 2.3, the definition of low-resource language and an overview of different types of languages in terms of the language data availability from the linguistic perspective are summarized.

When it comes to the research method, some cross-lingual retrieval methods are used in this work. Cross-lingual retrieval is one case of information retrieval. Retrieval-based methods are also popularly used in the NLP research. In Section 2.4, some basics of information retrieval are explained.

2.1 Pretrained Language Models (PLMs)

Language model (LM) is a core topic in the current research of computational linguistics and natural language processing (NLP). By applying different types of LMs, many various NLP tasks can be addressed at least to some extent. Every reform of LMs brings a major breakthrough in NLP.

The latest reform of LMs is the emergence and popularization of transformer-based language models based on a large amount of language corpora. Since 2018 alone, a series of LMs have continously emerged, such as BERT (Devlin et al., 2018), and its variants (Liu et al., 2019; Lan et al., 2019; Sanh et al., 2019), GPT-2 (Radford et al., 2019), XLNet (Yang et al., 2019b), GPT-3 (Brown et al., 2020) etc. One of the biggest trends in LMs has been the increasing size of the models. The size of LMs is measured by the number of parameters and size of training data. However, whether the development and deployment of even larger PLMs will bring some potential risks is questioned by some researchers in the NLP community (Bender et al., 2021).

2.1.1 Development of Language Models

Language Models are defined as models that assign possibilities to sequence of words (Jurafsky and Martin, 2000). Basically, LMs are used for language prediction. It is originally used to predict how probable a sentence is correct. Given a sentence S containing n words: $S = x_1x_2x_3 \cdots x_n$, the task of a language model $\mathcal{L}$ is to predict the probability of the sentences S. By applying the chain rule of probability, the probability of a sentence, i.e., a sequence of words, is converted to the product of the conditional probabilities of all words in the sequence given its previous context words, as equation (2.1) shows. It is still complex to compute the probability of a word when considering all its previous context. By applying the Markov assumption, it can be further simplified to compute the probability of a word only given the one word before it.

$$
\begin{aligned}
P(S) &= P(x_1x_2x_3 \cdots x_n) \\
&\overset{*}{=} P(x_1) \times P(x_2|x_1) \times P(x_3|x_1x_2) \times \cdots \times P(x_n|x_1 \cdots x_{n-1}) \\
&\overset{**}{=} P(x_1) \times P(x_2|x_1) \times P(x_3|x_2) \times \cdots \times P(x_n|x_{n-1})
\end{aligned}
\tag{2.1}
$$

** Applying the chain rule of probability*
*** Applying the Markov assumption*

Several different types of language models are developed in the past years, such as n-gram LMs, neural network based LMs and transformer-based LMs.

N-gram LM The n-gram model is the most basic LM type. N-gram means a sequence of n words. N-gram LMs refer to LMs that predict the probability of one word based on its previous $(n-1)$ words. The LM after applying the Markov assumption as shown in the equation (2.1) is a bigram LM. An n-gram is trained with a corpus. The bigram or n-gram probabilities can be estimated by the frequency of the bigrams or n-grams from the corpus.

The quality of a language model can be measured in two ways, i.e., extrinsic evaluation and intrinsic evaluation. Extrinsic evaluation is an end-to-end evaluation. To evaluate a LM in an extrinsic way, we can embed it in an specific task and measure how much the result is improved. An intrinsic evaluation metric is one that measures the quality of a model independent of any application. Perplexity is an intrinsic evaluation metric for LMs. Perplexity is defined as the normalized inverse probability of the test set, as equation (2.2) shows.

$$PP(W) = \sqrt[N]{\frac{1}{P(w_1, w_2, \cdots, w_N)}} \tag{2.2}$$

A good LM is supposed to assign the highest probability to the correct sentence. The sentences in the test set are regarded as correct, thus we expect a good LM to assign a high probability to the test set, which means the LM has a good understanding of how the language works. Since perplexity is the inverse probability of the test set normalized by the number of words in the test set, the lower the perplexity is, the higher quality the model has.

Neural Network Based LM The Neural network based LM is developed with the popularization of neural networks methods in NLP field. Neural network based LMs are designed to predict the next word in a sequence by its previous words based on the neural network structure, such as Recurrent Neural Network (RNN), Long Short Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) etc. Neural network based LMs adopt a more powerful method to represent words, i.e., the word embedding, a semantically meaningful vector. The concept of the word vector is clarified in Section 2.1.2

Figure 2.1 is an example of neural network method using an RNN structure. The hidden states $h_0, h_1, \cdots$ in the RNN layer transmit the information from all previous words to the next state. Therefore, the RNN language model can predict the next word considering all its previous context.

RNN architecture has a deadly limitation that restricts its ability to process long sequence data. In an RNN, at a given point in time j, the information about all past inputs is squeezed into one hidden state. For long sequence, the state containing the information of too many previous inputs becomes a bottleneck. The solution to this is the introduction of attention mechanism (Bahdanau et al., 2014), an architectural modification of the RNN encoder-decoder that allows the model to pay attention to those important past encoder states and ignore the irrelevant ones.

Transformer-Based LM The transformer-based model is based on the transformer architecture (Vaswani et al., 2017). Transformer is a network structure taking advantage of the self-attention mechanism. Different from attention mechanism by Bahdanau et al. (2014), transformer architecture consists of attention only. The transformer architecture is composed of two parts, Encoder and Decoder. Both parts contain modules that can be

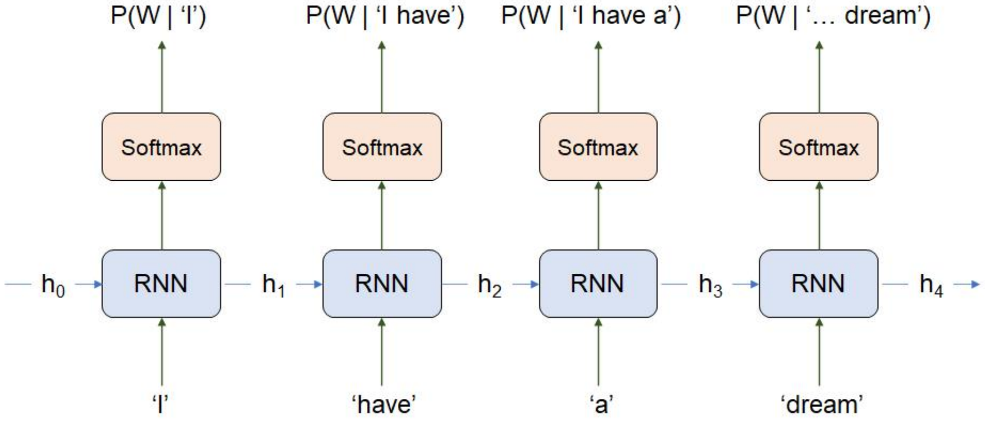


Figure 2.1: An example of RNN Language Model Structure

stacked on top of each other for multiple times. The core parts of each module are multi-head attention and feed forward layers. Transformer structure has powerful representation towards words. Transformer is the basic component of most pretrained language models.

2.1.2 Word Embeddings

Word embedding technology is the basic precondition to introduce the deep learning and neural network methods to language processing research, since deep learning is essentially a data processing method, where various types of data are fed into the neural networks as inputs and then a series of outputs are produced after layers of delivery. Based on the outputs of the neural networks, a variety of language tasks can be done, such as classification, sequence labeling, language generation etc. If we want the deep neural networks to solve NLP problems, the first step we should consider is to represent the language texts in a data form that can be directly processed by the neural networks. The word representation in the context of neural networks can be understood as the digitalization of words.

One-Hot Encoding An intuitive idea is to encode each word in a vocabulary with an ID. Such word vector method is called one-hot encoding, which means each word is represented by a vector only containing 0 and 1. Suppose the size of the vocabulary is $|V|$, then the length of each vector is $|V|$. Only the vector value on the position corresponding to the word ID is 1, all other positions have the value of 0. If we stack all word vectors according to their ID order, then we can get a diagonal matrix of $|V| \times |V|$ size used to represent the whole vocabulary. The values on the diagonal are 1.

One-hot encoding is a simple method to represent words, however, there are two major drawbacks. One is the explosion of parameters. The dimension of word vector is equal the vocabulary size $|V|$, which could even reach millions. Such parameter size is too large. The other problem lies that the vocabulary is represented by a diagonal matrix, which causes that all word vectors are orthogonal to each other, thus the notion of word similarity does not exist in the one-hot encoding. Therefore, a more effective approach of word representation is required to overcome these two issues. Firstly, we want the dimension of the new word vectors $|D|$ on a relatively low level, e.g., $50 \leq |D| \leq 1000$. Besides, the information of semantic similarity between words should be included in the word vectors and can be represented by the cosine between two vectors, as the (2.3) shows.

$$cos(\mathbf{w}^{(i)}, \mathbf{w}^{(j)}) = \frac{\mathbf{w}^{(i)T}\mathbf{w}^{(j)}}{||\mathbf{w}^{(i)}||_2 \cdot ||\mathbf{w}^{(j)}||_2} \tag{2.3}$$

Static Word Embedding The static word embedding satisfies both requirements mentioned above. The idea of word embeddings is based on a hypothesis from the distributional

Type	Model	Parameters	Dataset Size
Masked LM	BERT (Devlin et al., 2018)	3.40E+08	16GB
	RoBERTa (Large) (Liu et al., 2019)	3.55E+08	161GB
	DistilBERT (Sanh et al., 2019)	6.60E+07	16GB
	ALBERT (Lan et al., 2019)	2.23E+08	16GB
L2R LM	XLNet (Large) (Yang et al., 2019b)	3.40E+08	126GB
	GPT-3 Brown et al. (2020)	1.75E+11	570GB
Encoder-Decoder	BART (Large) (Lewis et al., 2019a)	4.00E+08	161GB
	T5-11B (Raffel et al., 2020)	1.10E+10	754GB

Table 2.1: An Overview of Different Types of PLMs

semantics, 'a word is characterized by the company it keeps' (Firth, 1957).

Word2vec model (Mikolov et al., 2013) is a classic word embedding model trained by negative sampling. In this method, word vectors are regarded as model parameters. The model is trained on the context information. There are two methods for negative sampling dependent on the objective functions, the skip-gram model and the Continuous-Bag-Of-Words (CBOW) model. In the skip-gram model, the training task is to predict the context words given the target word, while the task for CBOW model is opposite, predicting the target word given the context words.

FastText (Bojanowski et al., 2017) makes an improvement on the basis of word2vec model. One major limitation of word2vec model is that unknown words, which have not appeared in the training corpus, or out-of-vocabulary (OOV) words, which are outside the vocabulary, cannot be tackled by the word2vec embedding. FastText model addresses the problem by sampling character n-grams instead of tokens as word2vec does. FastText also computes the embeddings of n-grams of each word besides its original word embedding. In this way, the word vector of an unknown word can be represented by its subword vectors.

GloVe (Pennington et al., 2014) further improves the word2vec model by not only considering the local statistics of a corpus but also incorporateing the global statistics in that it uses the co-occurance matrix to derive semantic relationships of words.

Contextualized Word Embedding The contextualized word embedding is different from traditional word type embedding in that each token is assigned a representation that is a function of the entire input sentence. Contextualized word embeddings are derived from language models. ELMo (Peters et al., 2018) is an example of contextualized word embeddings learned from a bidirectional LM. ELMo embeddings combine a forward LM, a backward LM and a context-independent character-based token representation.

The first layer of a transformer-based language model is the embedding layer. The parameters of the embedding layer are also contextualized word embeddings learned from a LM.

2.1.3 Different Types of PLMs

The paradigm of pretrained language models is thriving in NLP field in recent years. With the help of increasingly higher computational power, researchers can train the deeper and more complex architectures language models with even more massive amounts of corpora. Substantial work has shown that PLMs can investigate linguistic features and general knowledge from the training corpora and encode them into their large-scale parameters.

PLMs are trained on different pretraining tasks. According to the different types of pretraining methods, PLMs can be classified as masked LMs, left-to-right LMs and Encoder-Decoder. Table 2.1 gives an overview of different types of PLMs and some typical examples for each type.

Masked LM The masked LM is an auto-encoder model, since they widely use the a bidirectional objective function, masked language modeling (MLM), to learn representation in the pretrtaining. MLM aims to predict the masked text pieces based on its surrounding context.

One representative masked LM is the BERT model (Devlin et al., 2018). BERT learns the bidirectional encoder representations from the transformer architecture. By introducing two training objectives, masked language modeling (MLM) and next sentence prediction (NSP), it can be pretrained on purely raw language data in a self-supervised pattern. The training objective of MLM is to predict the masked tokens given a sentence. The samples are generated by randomly replacing 15% of the tokens in the training corpora. The second training objective is to predict whether the second sentence follows the first sentence given two sentences. The negative examples are sampled by randomly selecting a sentence from the corpora. The pretraining corpora for BERT are from BooksCorpus and the English Wikipedia. The tokenization method of BERT is WordPiece (Schuster and Nakajima, 2012).

Other typical masked LMs include the variants of BERT models, ERNIE (Zhang et al., 2019) etc.

Left-to-right LM The left-to-right LM is an auto-regressive LMs, predicting the upcoming words or assign a probability $P(\mathbf{x})$ to a sequence of words $\mathbf{x} = x_1 \cdots x_n$. The likelihood $P(\mathbf{x})$ is factorized by using a chain-rule in a left-to-right fashion: $P(\mathbf{x}) = \prod_{t=1}^{T} p(x_t|\mathbf{x} < t)$. Auto-regressive LMs can only uni-directionally predict, broken down either forward or backward. XLNet (Yang et al., 2019b) is an example of left-to-right LMs. XLNet uses permutation language modeling as pretraining objective function. It is the factorization order that is permuted instead of the sequence order. By adopting PLM, bidirectionality is allowed while keeping auto-regressive objective.

Other representatives of modern left-to-right LMs include GPT-3 (Brown et al., 2020) etc.

Encoder-Decoder The encoder-decoder is a model that uses a type of LM to decode y conditioned on a separate encoder for text x with a fully-connected mask. The encoder and decoder do not share their parameters to each other. T5 model (Raffel et al., 2020), Text-to-text transfer transformer, is a complete encoder-decoder transformer architecture. T5 model reformulates all downstream NLP tasks as text-to-text tasks. Both dataset and parameter size rise up to a large-scale level, from BERT-size up to more than 750GB in dataset size and to 11 billion model parameters.

Besides, encoder-decoder LM structure is widely used in many other PLMs such as BART (Lewis et al., 2019a), MASS (Song et al., 2019) and their variants.

2.2 Multilingual Pretrained Language Models (MPLMs)

2.2.1 Motivation of MPLMs

The emergence of PLMs pretrained on massive amounts of unlabeled raw language data has influenced the research paradigm of NLP in recent years. With transfer learning, the PLMs have achieved impressive performance in many downstream NLP tasks even with amounts of annotated data. However, most of the research still focuses on the English text data, while low- and medium-resource languages benefit not enough from the development of PLMs. To alleviate the problem, multilingual pretrained language models, which are capable of processing more than one language with comparable performance, came into being. MPLMs have distinc adavantages over monolingual PLMs in terms that fewer models need to be pretrained an maintained. Besides, MPLMs possess cross-lingual transfer ability, and low- and medium-resource languages can thus benefit from it in many fields such as machine translation, zero-shot task transfer and typological research.

MPLMs are constructed based on the PLMs. On the basis of PLMs, we expand the monolingual training corpora to multilingual unlabeled corpora and map all language representations to the same semantic vector space. That is to say, multilingual word embeddings are created from the joint pretraining across different languages. The structure and pretraining objectives of MPLMs are the same as those of PLMs. Compare to PLMs, MPLMs have larger vocabulary size. For example, the the vocabulary size of the base version of BERT is 28,996, while the the vocabulary size of the base version of multilingual BERT (mBERT) is 119,547.

2.2.2 Popular MPLMs

mBERT Multilingual BERT (Devlin et al., 2018) can process the largest 100 languages in terms of the Wikipedia size and uses the corresponding Wikipedia corpora of the 100 languages as pretraining dataset. Like BERT, the pretraining tasks of mBERT are MLM and NLP too. mBERT is also pretrained in a self-supervised setting.

Considering that the sizes of Wikipedia of different languages vary a lot, the low-resource languages could be under-presented in the neural network models. What's more, the model could overfit a tiny Wikipedia for a particular language. Therefore, some sampling tactics are adopted when creating the pretraining dataset from the multilingual Wikipedia. Exponential smoothing is used to under-sample high-resource languages like English and to over-sample low-resource languages like Icelandic. What exponential does is change the the sampling probability distribution of each language. The original distribution is the same as the frequency of languages. For example, English corpora occupies 21% of the total corpora, and then the sampling probability for English is also 21%. After exponential smoothing, the frequency of all languages is exponentiated by a factor S, e.g. $S = 0.7$. We then normalize the exponentiated probabilities. In the original probability distribution, English is 1,000 times as probably sampled as Icelandic, the new distribution after smoothing makes English be sampled 100 times as probably as Icelandic.

Like BERT model, mBERT uses WordPiece (Schuster and Nakajima, 2012) to tokenize the words.

XLM XLM model (Lample and Conneau, 2019) is also a transformer-based MPLM. Similar to BERT, the pretraining task of XLM includes MLM too. Based on that, XLM also adds a second pretraining objective, Translation Language Modeling (TLM), which is used to reinforce the cross-lingual representation between different languages. For the pretraining with the MLM objective, XLM uses Wikipedia as the pretraining dataset. For the TLM pretraining task, XLM selects parallel corpora according to different languages.

The basic XLM model includes 15 languages, with two extended versions, each containing 17 and 100 languages. The two extended versions of XLM are only pretrained with the MLM task, without using TLM. XLM uses BytePair Encoding (BPE) (Gage, 1994; Sennrich et al., 2015) for tokenization. The vocabulary size of the extended version is approximately 200,000.

XLM-R XLM-R (Conneau et al., 2019) is the multilingual version of RoBERTa (Liu et al., 2019). XLM-R have not used TLM as a pretraining objective as the original XLM does, but it is pretrained in the same way as RoBERTa on 2.5TB filtered web-crawled data provided by CommonCrawl, containing 100 languages. XLM-R has a vocabulary size up to 250,000, compared to the vocabulary size of 50,000 of the original RoBERTa.

M2M100 M2M100 (Fan et al., 2021) is a multilingual translation model with en encoder-decoder (seq2seq) structure, used for many-to-many multilingual translations. This model is capable of translating between any two of 100 languages with $100 \times 99 = 9900$ translation directions in total. When using M2M100 model for translation, we need to use the ID of the target language as the first token to encode.

mBART-50 mBART (Liu et al., 2020) is the multilingual version of BART (Lewis et al., 2019a), a sequence-to-sequence denoising auto-encoder pretrained on large-scale monolingual corpora in many languages using the BART objective. mBART uses the method for pretraining a complete sequence-to-sequence model by denoising full texts in multiple languages. mBART-50 supports the translation between 50 languages. This model proves that multilingual translation models can be realized by using the multilingual finetuning method. Different from normal finetuning in just one translation direction, pretrained mBART-50 is able to finetune in multiple directions at the same time.

mBART uses multilingual denoising pretraining. At first, all monolingual corpora are concatenated to be one dataset $D = \{D_1, D_2, \cdots, D_n\}$, where D_i is the monolingual corpus of the language i. The texts of source languages are labeled by noise in two ways: sentence permutation and word-span masking. The objective of multilingual denoising pretraining is to reconstruct the original texts. Like M2M100, mBART-50 uses a special token before the input text to mark different languages as well.

2.3 Language Resources

The success of modern NLP methods on the basis of large amounts of labeled and unlabeled corpora mainly benefits those languages with high digital resources, which only take up a very small number of the over 7,000 languages of the world. The majority of the world's languages do not have sufficient digital resources and thus cannot benefit from recent progress in NLP. The emergence of MPLMs alleviates the problem to some extent, but is still far from addressing it. MPLMs trained on up to a hundred languages have shown surprisingly good cross-lingual transfer performance on some NLP tasks, even without explicit cross-lingual signals (Wu and Dredze, 2019). However, the good performance only works well for those languages covered by mBERT with relatively high resources. Wu and Dredze (2020) finds that not all languages are represented equal in MPLMs. With small monolingual corpus, MPLMs do not learn high-quality language representations. Apart from this, the pretraining of MPLMs is highly dependent on the availability of monolingual corpora, It means that most world's languages, which means they can by no means covered by the MPLMs.

In this subsection, an overview of the distribution of language resources in the world and how it affects the NLP research are presented.

2.3.1 Languages of Different Resources

The number of the languages in the world is dynamic, since some old languages are dying and new languages are emerging. Nowadays, there are more than 7,000 known languages in the world. However, the distribution of language resources varies greatly across different languages. Basically, the availability of language resources decides how much a language can benefit from the modern data-driven NLP methods.

MPLMs like mBERT is capable of covering around 100 languages, which barely occupy 1% of the languages. MPLMs have to be pretrained on large amounts of unlabeled language data. However, Wikipedia and CommonCrawl, the two most commonly used resources, provide textual data for only 316[1] and 160[2] languages respectively, only 4% of all languages. Bible is a resource that is available for the most number of languages. 1600 languages of Bible comprise the largest parallel corpora in terms of the language variety. The Bible has a language coverage of 23%, but still more than 70% of the world's languages do not even have any unlabeled digital data at all. However, the endeavor of linguists in studying and documenting the under-represented languages makes the bilingual lexicon or word lists for 70%[3] of the languages.

[1] https://en.wikipedia.org/wiki/List$_of_Wikipedias$

[2] https://commoncrawl.github.io/cc-crawl-statistics/plots/languages

[3] https://vocab.panlex.org/

Joshi et al. (2020) classified the 7,000 languages of the world into 6 kinds based on their digital status and richness in the context of data availability. They measured the language data richness by two features: the number of unlabeled resources and the number of labeled resources.

The category containing the languages with the fewest resources is called the 'left-behinds'. This kind of languages have virtually no unlabeled data to use and thus have been ignored in the wave of language technologies. With some amount of unlabeled data, the 'scraping-bys' are slightly better, but still needs more efforts to collect labeled data for them. With a small set of labeled datasets collected by the researchers, the languages in the category, the 'hopefuls', still struggles to stay alive in the digital world. With a strong presence in the world of Internet, the 'rising stars' have utilized the boosting of unsupervised pre-training, however, the lack of efforts in labeled data collection restricts their further study. The 'underdogs' have large amounts of unlabeled data, comparable to the languages possessed by the top category, and are only left behind in the amount of labeled data. The top group of languages in the context of resources is called the 'winners'. With a dominant online presence, they have incorporated massive industrial and government investments in the development of resources and technologies. They promote the breakthrough of the state-of-the-art methods and benefit most from them. They are the quintessential rich-resource languages.

2.3.2 Language Distribution in MPLMs

This work focuses on the languages in the MPLMs. The languages in the MPLMs can be roughly categorized into three classes: Low-, medium- and high-resource languages.

Low-resource languages have received much concern in the recent studies of NLP. Singh (2008) understood low-resource languages as resource scare, less studied, less computerized and less privileged languages. Tsvetkov (2017) defined low-resource languages as languages lacking large monolingual or parallel corpora and/or manually crafted linguistic resources sufficient for building statistical NLP applications. Agić et al. (2016) have further explained in their work for truly low-resource languages, i.e., languages with no supporting tools or resources for segmentation, POS tagging, or dependency parsing. As mentioned above, this work studies low-resource language by using MPLMs as tools, so the classification of low-, medium- and high-resource languages are categorized according to their distribution in MPLMs' pretraining corpora.

Table 2.2 displays the list of 99 languages with the largest Wikipedia size and to which language family the languages belong. Since mBERT is pretrained on the largest Wikipedias of top 100 languages (Devlin et al., 2018), this table also shows the distribution of languages in the pretraining dataset of mBERT. The table lists the languages according to the order of their Wikipedia size. The first column is the full name of the language, while the second column marks the corresponding ISO code. The information of language family is provided in the third column. The last column notes the size range of the Wikipedia.

The language family reflects the genetic relationship and distance between different languages. It receives attention in the multilingual NLP, because the similarities of language structure and linguistic features could influence the cross-lingual performance (Lauscher et al., 2020). Regarding the languages covered by mBERT, around 60% of the languages are from the Indo-European family. Others are from Altaic, Sino-Tibetan, Austronesian, Uralic, Dravidian, Afro-Asiatic, Niger-Congo and Caucasian language families. It is notable that two artificial languages created by humans are also included in the mBERT. They are Volapük and Ido.

In this study, languages with the Wikipedia size larger than 1.414GB are categorized as high-resource languages, such as English, Russian, German, Chinese etc. Languages with the Wikipedia size larger than 0.1777GB and less than 1.414GB are treated as medium-resource languages, such as Ukrainian, Vietnamese, Turkish etc. All other languages with the Wikipedia size less than 0.177GB are low-resource languages, such as Urdu, Telugu,

Language	ISO	Family	Size Range (GB)
English	en	Indo-European	[11.314, 22.627]
Russian	ru	Indo-European	[2.828, 5.657]
French	fr	Indo-European	
Spanish	es	Indo-European	
German	de	Indo-European	
Chinese	zh	Sino-Tibetan	[1.414, 2.828]
Portuguese	pt	Indo-European	
Polish	pl	Indo-European	
Japanese	ja	Altaic	
Italian	it	Indo-European	
Cebuano	ceb	Austronesian	
Ukrainian	uk	Indo-European	[0.707, 1.414]
Swedish	sv	Indo-European	
Dutch	nl	Indo-European	
Hungarian	hu	Uralic	
Czech	cs	Indo-European	
Catalan	ca	Indo-European	
Arabic	ar	Afro-Asiatic	
Vietnamese	vi	Austroasiatic	[0.354, 0.707]
Turkish	tr	Altaic	
Serbian	sr	Indo-European	
Romanian	ro	Indo-European	
Norwegian	no	Indo-European	
Korean	ko	Altaic	
Indonesian	id	Austronesian	
Hebrew	he	Afro-Asiatic	
Finnish	fi	Uralic	
Persian	fa	Indo-European	
Waray Waray	war	Austronesian	[0.177, 0.354]
Thai	th	Tai-Kadai	
Slovenian	sl	Indo-European	
Slovak	sk	Indo-European	
Serbo Croatian	sh	Indo-European	
Malay	ms	Austronesian	
Armenian	hy	Indo-European	
Croatian	hr	Indo-European	
Galician	gl	Indo-European	
Estonian	et	Uralic	
Greek	el	Indo-European	
Danish	da	Indo-European	
Bulgarian	bg	Indo-European	
Belarusian	be	Indo-European	
Asturian	ast	Indo-European	
Urdu	ur	Indo-European	[0.088, 0.177]
Telugu	te	Dravidian	
Tamil	ta	Dravidian	
Norwegian Nynorsk	nn	Indo-European	
Malayalam	ml	Dravidian	
Mecedonian	mk	Indo-European	
Latvian	lv	Indo-European	
Lituanian	lt	Indo-European	[0.088, 0.177]
Kazakh	kk	Altaic	
Georgian	ka	Caucasian	
Hindi	hi	Indo-European	
Basque	eu	Language Isolate	
Bosnian	bs	Indo-European	
Bengali	bn	Indo-European	
Azerbaijani	az	Altaic	
Uzbek	uz	Altaic	[0.044, 0.088]
Tatar	tt	Altaic	
Tagalog	tl	Austronesian	
Albanian	sq	Indo-European	
Scots	sco	Indo-European	
Occitan	oc	Indo-European	
Marathi	mr	Indo-European	
Latin	la	Indo-European	
Kannada	kn	Dravidian	
Welsh	cy	Indo-European	
Bashkir	ba	Altaic	
Afrikaans	af	Indo-European	
Tajik	tg	Indo-European	[0.022, 0.044]
Swahili	sw	Niger-Congo	
Western Punjabi	pnb	Indo-European	
Punjabi	pa	Indo-European	
Nepali	ne	Indo-European	
Low Saxon	nds	Indo-European	
Burmese	my	Sino-Tibetan	
Mongolian	mn	Altaic	
Lombard	lb	Indo-European	
Kirghiz	ky	Altaic	
Javanese	jv	Austronesian	
Icelandic	is	Indo-European	
Gujarati	gu	Indo-European	
Irish	ga	Indo-European	
West Frisian	fy	Indo-European	
Chechen	ce	Caucasian	
Breton	br	Indo-European	
Bavarian	bar	Indo-European	
Aragonese	an	Indo-European	
Volapük	vo	Artificial	[0.011, 0.022]
Sudanese	su	Afro-Asiatic	
Minangkabau	min	Austronesian	
Malagasy	mg	Austronesian	
Luxembourgish	lmo	Indo-European	
Chuvash	cv	Altaic	
Yoruba	yo	Niger-Congo	[0.006, 0.011]
Sicilian	scn	Indo-European	
Pietmontese	pms	Indo-European	
Ido	io	Artificial	

Table 2.2: List of the 99 Languages with the largest Wikipedia size and the language family they belong to.

Swahili etc.

2.4 Information Retrieval

This work aims to improve the low-resource languages' performance on NLP tasks with the help of cross-lingual information. A central step for that is the retrieval of cross-lingual information. In NLP research, it is common to leverage information retrieval methods to collect external knowledge and resources to solve different NLP tasks. Retrieval-based methods can be classified into two types dependent on the representations for retriever. The first type retriever uses sparse representation based on bag-of-word (BOW) (Chen et al., 2017). The second uses dense representation from neural networks Karpukhin et al. (2020).

2.4.1 Retrieval by Sparse Representation

The sparse representation method is based on BOW and can be easily applied to a general large-scale search and open domain question answering (Chen et al., 2017). Retriever with sparse representation usually computes rule-based scores used for document ranking. This can be seen as representing the query and context in high-dimensional and sparse vectors with weighting.

For full-text search collections, a retriever should pay attention to term frequency (tf) and document length. BM25 (Schütze et al., 2008) score has been used quite widely and quite successfully across a range of collections and search tasks. BM25 is a probabilistic model that incorporates tf and length normalization, modified on the basis of the Binary Independence Model (BIM) (Yu and Salton, 1976) score for document d only dependent on the inverse document frequency (idf) weighting of the query terms present in the document:

$$RSV_d = \sum_{t \in q \cap d} log \frac{N}{df_t} \tag{2.4}$$

BM25 score improves the idf term $\frac{N}{df}$ in the equatio (2.4) by factoring in term frequency and document length, as the equation (2.5) shows.

$$RSV_d = \sum_{t \in q \cap d} log \frac{N}{df_t} \cdot \frac{(k_1 + 1) tf_{td}}{k_1((1 - b) + b \times (L_d / L_{ave})) + tf_{td}} \tag{2.5}$$

- tf_{td}: term frequency in document d
- L_d (L_{ave}): Length of document d (average document length in the whole collection)
- k_1: tuning parameter controlling scaling of term frequency
- b: tuning parameter controlling the scaling by document length

Sparse representation has the advantage of generalization and efficiency, and is thus apt to search in a large-scale document pool.

2.4.2 Retrieval by Dense Representation

Representation can be in a dense form by latent semantic encoding as well. Dense representations derive from encoders of neural network structure, pretrained on task-specific data. With the popularization of the transformer-based models in recent years, dense representation based retrieval has been the most widely explored area, compared to traditional sparse representation based retrieval. Dense representation is designed to be a good complement to sparse representation. For example, for synonyms or paraphrases sharing no common tokens with each other, sparse representation based retriever cannot match them properly as we hope. In dense representation based retrieval, however, synonyms and paraphrases are mapped to in a vector space, where their vectors are close to each other, so they are more probably matched. Therefore, retrieval methods by dense representation can achieve better recall performance than sparse representation on different tasks, such as open domain question answering (Karpukhin et al., 2020).

However, dense representation retrieval has two drawbacks that constrain its usage to be generalized and efficient. One is that learning a good dense vector representation needs a large number of labeled pairs of question and contexts. It needs the parallel data for model training on some specific tasks. Besides, limit to the transformer structure, retriever with dense representations cannot process very long documents. The maximum length that the retriever can process is restricted by the maximum length of sequence previously set by the transformer.

3 Related Work

This chapter introduces previous studies on which this work is based. This work concentrates on the performance improvement of MPLMs on the low-resource languages by exploiting the multilinguality of MPLMs and combining several recently popular methods, such as zero-shot learning and cross-lingual retrieval. Section 3.1 introduces the recent study on the explanations of multilinguality equipped by MPLMs. Section 3.2 summarizes how previous research applies MPLMs to cross-lingual transfer learning. In section 3.3, prompt-based learning, a novel but popular concept used in large-scale PLMs, is explained. In the last section 3.4, the development of transformers on sentence and document level useful for information retrieval are presented.

3.1 Multilinguality

Multilinguality is the property and ability of multilingual representation. It can be understood that the words from different languages are mapped into the same vector space and can be compared to each other directly. For example, if the German word 'Hund' and the English word 'dog' are represented by a multilingual model, then they should be close to each other in the vector representation space of the model. It has been shown that MPLMs learn multilingual representation by joint pretraining on large amounts of multilingual corpora. In recent studies, MPLMs have shown its multilinguality, i.e., high quality multilingual representations (Lauscher et al., 2020). However, it is surprising that MPLMs learn such multilingual representation without using any explicit signals that link different languages in the pretraining. The MPLMs are told no external information of how different languages are linked, but they still show multilinguality after pretraining. Therefore, some research on the explanation and analysis of the MPLMs' multilinguality has been conducted by the NLP community recently.

Singh et al. (2019) reveals that mBERT partitions representations for each language rather than using a common, shared, interlingual space as expected. They analyze mBERT using projectiob weighted canocial correlation analysis (PWCCA) (Hotelling, 1992), which is used to investigate the relationships between two sets of random variables and analyze the representations of the same data points from different models in a way that is invariant to affine transformations (Morcos et al., 2018). PWCCA is thus particularly suitable for the analysis of neural networks. By using PWCCA similarity scores, mBERT can be partitioned, which suggests that mBERT does not represent semantically similar data points closer to each other in a common space. By using unweighted pair group method with arithmetic mean (UPGMA), a simple agglomerative hierachical clustering method (Sokal, 1958), the researchers generate a phylogentic tree from the representations from Layer 6 of mBERT, which closely resembles the linguistic language tree reflecting the relationships and evolution of human languages. At deeper layers, the partition is magnified, suggesting that the mBERT abstract the semantic contents regarding features related to natural differences and similarities between languages. BERT employs WordPiece (Schuster and Nakajima, 2012) for tokenization rather than character- or word-level tokenizations. Their work shows that the subword tokenization is a motivation for mBERT to discover these linguistic and evolutionary relationships between languages.

Artetxe et al. (2019) prove that neither joint pretraining nor shared vocabulary is essential for mBERT's multilinguality by designing an alternative approach that transfers a monolingual model to new languages at the lexical level. They first train a transformer-based masked language model on one language, and transfer it to a new language by learning a new embedding matrix without using a shared vocabulary or joint training. However, the result is competitive with mBERT, contradicting the hypothesis that shared subword vocabulary and joint training contribute to the multilinguality of MPLMs.

Wang et al. (2019) extensively research different potentially essential components for the MPLMs' multilinguality by experiments and further prove that lexcial overlap between languages make neligible contributions to the multilinguality. They study the essential elements for MPLMs' multilinguality in three dimensions: linguistic properties (word-piece overlap, word-ordering similarity, word-frequency similarity and structural similarity), model architecture (model depth, multi-head attention, number of parameters etc.) and learning objective (NSP, language identity marker, types of tokenization). Their comprehensive experiments conclude that word-piece overlap and multi-head attention are not significant for the multilinguality, while structural similarity between languages and the depth of MPLM are crucial.

Wu et al. (2019) conduct experiments to validate four factors that possibly play important roles in MPLMs' multilinguality: domain similarity, shared vocabulary, shared parameters and language similarity. They find that shared vocabulary across monolingual corpora and domain similarity of the corpora are not important. Shared parameters in the top layers of MPLMs are required for the cross-lingual ability of MPLMs. They further clarify the results by showing that the representations of monolingual BERT in different languages are in isomorphic spaces and can be aligned post-hoc effectively. They believe the MPLMs can take advantage of the universal latent symmetries in the learned embedding spaces of different languages and aligned them during the joint training process automatically.

Dufter and Schütze (2020) investigate 4 architectural properties (overparameterization, shared special tokens, shared position embeddings and random word replacement) and 2 linguistic properties (word order and comparability of corpora) that could be the reasons for the multilinguality in a clean laboratory setting. Their experiment results show that the limited number of parameters, shared special tokens, shared position embeddings and random masking strategy contribute to multilinguality. Different from previous work, Dufter and Schütze (2020) do not use extrinsic metrics based on task performance of the model to evaluate the multilinguality. Instead, they create a comprehensive metric, the multilinguality score, to measure the model's multilinguality directly.

Deshpande et al. (2021) focus on exploring the influence of linguistic properties on multilinguality by performing large-scale experiments. Contrary to previous work, they find that the subword overlap significantly affects the multilinguality when languages have different types of word order. Besides, the word embedding alignment between languages strongly correlates to the multilinguality.

Curse of Multilinguality Conneau et al. (2019) observe the phenomenon with MPLMs that for a fixed model capacity, the cross-lingual transfer performance improves when adding more pretraining languages only up to a certain point. After that, adding more languages to pretraining degrades the performance. This phenomenon is termed the curse of multilinguality". It can be alleviated by increasing the model capacity (Artetxe et al., 2019). However, Dufter and Schütze (2020) point out that too many model parameters will harm multilinguality and there exists a trad-off between good generalization and high degree of multilinguality in MPLMs.

3.2 Cross-Lingual Transfer Learning

Transfer learning studies how machine learning models can be transferred to data outside of their training distribution (Pan and Yang, 2009), such as across different tasks, domains and languages. The idea of transfer learning is motivated by the cost of linguistic annotation and the large number of structurally different NLP tasks. The data scarcity in

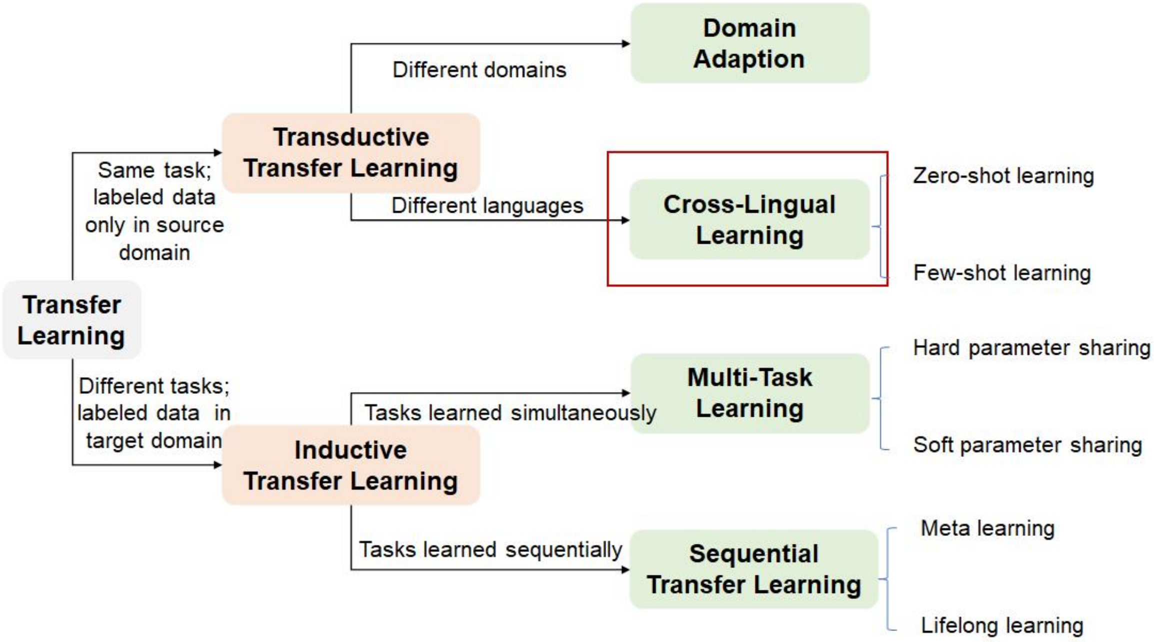

Transfer Learning
Same task; labeled data only in source domain
Transductive Transfer Learning
Different domains
Domain Adaption
Different languages
Cross-Lingual Learning
Zero-shot learning
Few-shot learning
Different tasks; labeled data in target domain
Inductive Transfer Learning
Tasks learned simultaneously
Multi-Task Learning
Hard parameter sharing
Soft parameter sharing
Tasks learned sequentially
Sequential Transfer Learning
Meta learning
Lifelong learning

Figure 3.1: Cross-lingual Transfer Learning in the Taxonomy of Transfer Learning (Ruder, 2019)

low-resource languages renders the need for effective cross-lingual transfer methods. Cross-lingual transfer learning can be understood as strategies to exploit abundant labeled data from high-resource languages to perform NLP tasks for low-resource languages. Figure 3.1 shows the position of cross-lingual transfer learning in the taxonomy of transfer learning methods. In zero-shot scenario of cross-lingual transfer, no single annotated data example from the target language is available. If a few labeled examples can be used, then it is few-shot scenario.

3.2.1 Empirical Study on Performance of Cross-Lingual Transfer

Cross-lingual word embeddings can used for cross-lingual transfer (Ruder et al., 2019). In more recent study, pretrained multilingual text encoders have gradually become a default paradigm for cross-lingual transfer learning. A batch of empirical studies on cross-lingual transfer with MPLMs have been conducted in recent year.

Pires et al. (2019) performs cross-lingual transfer probing experiments on named entity recognition (NER) and part-of-speech tagging (POS) tasks. Their results show that cross-lingual transfer with mBERT is effective for NER and POS between languages of different scripts with zero lexical overlap, while for typologically similar languages, transfer works better. Wu and Dredze (2019) experiment cross-lingual transfer with a broader range, on 5 tasks, such as text classification, dependency parsing, NLI etc., covering 39 languages.

With the popularization of cross-lingual transfer research, Hu et al. (2020) introduce XTREME, a benchmark for evaluating cross-lingual transfer performance with MPLMs containing 9 tasks and each task covers a subset of 40 languages. The tasks are categorized into 4 types: sentence classification (cross-lingual natural language inference (Conneau et al., 2018) and cross-lingual paraphrase adversaries from word scrambling (Yang et al., 2019a)), structured prediction (POS tagging and NER), sentence retrieval (parallel sentences extraction from a comparable corpus and nearest sentence retrieval) and question answering (cross-lingual questions answering dataset (Artetxe and Schwenk, 2019), multilingual question asnwering dataset (Lewis et al., 2019b) and typologically diverse question answering dataset (Clark et al., 2020)).

3.2.2 Limitations of Cross-Lingual Transfer

In spite of the impressive performance of zero-shot cross-lingual transfer method with MPLMs on many tasks, the limitations of the method have recently accumulated increasing concern from the NLP community.

Wu and Dredze (2020) compare the task performance of mBERT between low- and high-resource languages and show that mBERT does much worse for low-resource languages. Lauscher et al. (2020) further prove that not only the size of pretraining corpora of a language but also the linguistic similarity between the source and target languages influences the transfer performance.

3.2.3 Improvement of Cross-Lingual Transfer

Some approaches are proposed in order to improve the cross-lingual performance of MPLMs on the low-resource target languages.

Pfeiffer et al. (2020) prove that continued pretraining on monolingual text (Howard and Ruder, 2018) in the target language using a masked language modeling (MLM) objective can effectively adapts MPLMs to the target language. Another approach attempts to expand the labeled data of low-resource languages for finetuning by introducing a machine translation system. Using such a machine translation system, we can translate the labeled data in the source language into target language data and finetune the pretrained MPLM on both source and target language data (Jundi and Lapesa, 2022). Lauscher et al. (2020) demonstrates that exploiting inexpensive labeled data of low-resource languages brings a surprising effectiveness across the board. They thus argue that efforts should be done to overcome the zero-shot conditions. In a most recent study, Wang et al. (2022b) propose a new idea to expand MPLMs to more low-resource languages through the use of bilingual lexicons annotated and documented by linguists.

3.3 Prompt-Based Learning

Prompt-based learning is a novel NLP paradigm with the development of even larger-scale PLMs and dubbed as the second sea change in NLP following the pretraining-finetuning paradigm (Liu et al., 2021). Unlike traditional supervised learning, which trains model to take in an input $\mathbf{x}$ and predict an output $\mathbf{y}$ as $P(\mathbf{y}|\mathbf{x})$, prompt-based learning employs language models, which predict the probability of text (see section 2.1.1, to perform various NLP tasks directly. To achieve that, we need to reformulate the form of input $\mathbf{x}$ to cloze-style or text-to-text query to which a PLM gives an answer.

For example, in the sentiment analysis task, the original input text "*This product is amazing.*" is modified by using a template defined by the prompting function $f_{prompt}(\mathbf{x})$ into a textual string prompt $\mathbf{x'}$, as the equation (3.1) shows. In this example, the original input is reformulated as "*This product is amazing. In summary, it is a* $[Z]$ *product.*". The label for this example should be "1" (positive). With the verbalizer that converts a label to a word (Schick and Schütze, 2020a), as the equation (3.2) shows, the label "1" can be converted to the word "great" and filled into the prompt. Prompt can also be constructed with more information, such as task description (Radford et al., 2019) and few-shot examples (Brown et al., 2020).

$$\mathbf{x'} = f_{prompt}(\mathbf{x}) \tag{3.1}$$

$$\mathbf{z} = v(\mathbf{y}) \tag{3.2}$$

3.3.1 Development of Prompt-Based Methods

Prompting methods derive from the development of super size of PLMs. Radford et al. (2019) shows the effectiveness of task description to the success of zero-shot task transfer with GPT-2. Brown et al. (2020) use GPT-3 to perform NLP tasks from only a few examples. This approach does not require to update the model parameters for making predictions and dubbed as "in-context learning", which means the model makes predictions by learning from a natural language prompt describing the language task or learning from examples. GPT-3 is capable of achieving strong performance on many NLP tasks, including some complicated reasoning and generation tasks with the in-context learning prompting method.

However, the model size of GPT-3 is prohibitively large with 175B parameters. Schick and Schütze (2020a) prove that prompting approaches are suitable for relatively smaller models, like RoBERTa and ALBERT, by reformulating the input examples as cloze-style phrases such that the PLMs understand a given task. They introduce the PET method, combining prompting methods with gradient-based optimization, and show that PLMs that are much smaller and "greener" can obtain performance similar to GPT-3 and even outperforms GPT-3 on SuperGlue with only 32 training examples. (Schick and Schütze, 2020c). Apart from classification tasks, Schick and Schütze (2020b) further apply the combined methods to generation tasks, such as tex summarization and headline generation.

3.3.2 Prompting Engineering

The prompting methods using natural language to describe NLP tasks is termed as discrete prompting, a.k.a. hard prompting. The prompts used in GPT-3 model and PET method are human-designed. Besides manual template engineering, there are also some methods to automate the template design process. Gao et al. (2020) leverage the seq2seq pretrained model T5 (Raffel et al., 2020) to search template and generate prompts directly. Shin et al. (2020) use downstream training sample to automatically search for template tokens. Jiang et al. (2020) use data mining based methods to automatically find templates for a set of training inputs and outputs from a large text corpus. Paraphrase-based approach is a kind of semiautomatic prompt learning method, which paraphrases an existing seed prompt into a set of candidate prompts (Yuan et al., 2021; Haviv et al., 2021; Zhong et al., 2021).

Prompting can also be performed directly in the embedding space of the model. Such prompting learning the prompt in a continuous embedding space with stochastic gradient descent (SGD) is continuous prompting, a.k.a. soft prompting. Compared to discrete prompts, continuous prompts elicit more knowledge from PLMs (Qin and Eisner, 2021). Prefix-tuning, freezing the parameters of PLMs, solves generation tasks with higher parameter efficiency (Li and Liang, 2021).

3.3.3 Advanced Studies on Prompting Methods

Integrating the idea of prompting with the methods from other fields could facilitate better performance. Liu et al. (2022) integrate unlabeled data into prompt-based learning, attempting to exploit the large amount of unlabeled data to improve the zero-shot performance of PLMs without parameter updating. Their work combines the retrieval-based methods with prompting and provides a new window of prompting research. Motivated by prompting engineering, Wang et al. (2022a) modifies the traditional supervised learning process by retrieving similar information from the labeled training dataset for each input and concatenate it with the retrieved content.

In spite of the extraordinary progress in zero- and few-shot learning with prompt-based methods, there are also some discussions and doubts on the mechanism of prompting methods. Some research focuses on how prompting improves a PLM's performance. Webson and Pavlick (2022) argues that prompt-based models cannot really understand the meaning of their prompts by experimenting with manually designing misleading and irrelevant prompts. Some researchers propose that adding some explanations to prompts can improve the prompt-based learning performance (Lampinen et al., 2022; Kojima et al., 2022).

The work on prompting methods with MPLMs to cross-lingual transfer and low-resource languages still remains scarce. Zhao and Schütze (2021) apply discrete and soft prompting methods to XNLI task with MPLMs and show that prompting performs better than finetuning in few-shot crosslingual transfer and in-language training of multilingual natural language inference. Huang et al. (2022) proposed a novel method that uses a unified prompt for all languages in the zero-shot cross-lingual setting with MPLMs.

3.4 Sentence Transformers

Words can be represented by word embedding and directly applied to some NLP tasks or taken by language models as inputs. Likewise, mapping sentences or short text paragraphs to a dense vector space, such that similar sentences are close, also has extensive applications.

3.4.1 Representation of Sentences

A simple method is to derive sentence embeddings from word embeddings. A sentence can be represented by the average embedding of tokens within the sentence. The word embeddings can be either static, like GloVe embeddings (Pennington et al., 2014), or contextual embeddings from PLMs. The most commonly used approach is to pass single sentences through BERT and then to derive a fixed size of vector by either averaging the BERT output layer embeddings or by using the output of the special $[CLS]$ token. For example, May et al. (2019) uses this method to measure social biases. Similaryly, Zhang et al. (2020) sums the token similarities in a sentence to evaluate the similarity between the candidate sentence and reference sentence.

Some other methods are proposed to represent sentences. Kiros et al. (2015) train an encoder-decoder model to reconstruct the surrounding sentences of an encoded passage and map sentences sharing semantic and syntactic properties to similar vector representations. Conneau et al. (2017) use supervised natural language inference datasets to train universal sentence representations from a siamese BiLSTM network with max-pooling over the output. After that, Cer et al. (2018) trains a transformer-based model, the Universal Sentence Encoder. In recent study, by modifying the network of transformer-based PLMs, sentence transformers can be trained based on the PLMs in a simple way. Reimers and Gurevych (2019) apply siamese and triplet network structures to BERT and RoBERTa to derive semantically meaningful sentence embeddings that can be compared using cosine-similarity and achieves the state-of-the-art performance on semantic textual similarity (STS) task and other transfer learning tasks. They then apply the method to other PLMs and have developed a variety of sentence transformers.

3.4.2 Multilingual Sentence Embeddings

Like many cases in other NLP fields, the imbalanced distribution of language resources exits in sentence embeddings as well. Most existing sentence embeddings models are monolingual, usually only for English. Therefore, multilingual sentence embeddings using transfer learning or knowledge distillation methods can benefit more languages.

Chidambaram et al. (2018) trained the Multilingual Universal Sentence Encoder (mU-SE) in a multi-task setup on SNLI dataset (Bowman et al., 2015) and on a web-crawled question-answering pairs dataset consisting of more than a billion pairs. Reimers and Gurevych (2020) employ the approach of multilingual knowledge distillation to train multilingual sentence transformers based on sentence transformers. The MPLM XLM-R serves as the student model $\hat{M}$ and the sentence BERT as the teacher model M. A set of parallel sentences of source language and translated target language, $((s_1, t_1) \cdots (s_n, t_n))$ with t_i the translation of s_i, are required for training the student model, such that $\hat{M}(s_i) \approx M(x_i)$

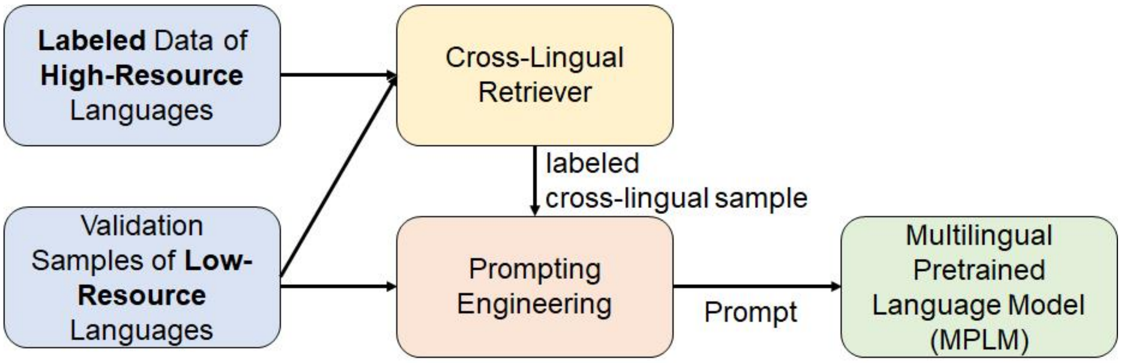
and $\hat{M}(t_i) \approx M(s_i)$ using mean squared loss (MSE) as loss function. This method is demonstrated to be effective for over 50 languages from different language families and can be extended to more languages easily.

4 Research Method

This chapter will introduce the details of our proposed methods. Section 4.1 presents the overview of two pipelines of the methods. The following sections introduce the detailed modules of the pipeline. Section 4.2 the sentence pool of high-resource languages used for retrieval. Section 4.3 is about the cross-lingual retrieval model used to obtain the semantically similar contents from the high-resource languages for the low-resource input. Section In the section 4.4, the prompting methods used to integrate the retrieved cross-lingual information are presented.

4.1 Pipelines

This work aims to improve the performance of MPLMs on low-resource languages under a zero-shot setting[1] by leveraging retrieved cross-lingual contents from high-resource languages. Dependent on whether the high-resource language data is labeled or not, two pipelines are designed.



Labeled Data of **High-Resource** Languages
Validation Samples of **Low-Resource** Languages
Cross-Lingual Retriever
labeled cross-lingual sample
Prompting Engineering
Prompt
Multilingual Pretrained Language Model (MPLM)

Figure 4.1: Pipeline of zero-shot on low-resource languages by cross-lingual retrieval from **labeled** high-resource language datasets.

Figure 4.1 shows the pipeline of zero-shot on low-resource languages (LRLs) by cross-lingual retrieval from labeled high-resource language (HRL) datasets. This pipeline starts from a labeled HRL corpora. The validation sample of a low-resource language is firstly fed into a cross-lingual retriever as a query and the retriever returns the most similar sentences from the high-resource labeled pool to the query. Then the low-resource input sample together with the retrieved cross-lingual sample is prompted by a specially designed prompting template[2]. At last, the MPLM takes the prompt created in the prompting engineering and makes predictions related to the task.

Table 4.1 shows an example of a low-resource input sample through the pipeline. The input sample is a Telugu sentence, which means "*absolutely does what was advertised!*" in English. The Telugu sample is firstly taken by the cross-lingual retriever as query. The retriever returns the most semantically similar samples of high-resource language, e.g. "*Great! Works as stated.*" together with its label "1", representing "positive". Then the input sample and retrieved sample are prompted in the prompting engineering. Prompting pattern is a template to reformulate the input, e.g. "[X] *In summary, the product was*

[1] Zero-shot setting in this work refers in particular to the low-resource languages and is not applicable to high-resource languages.

[2] Prompting templates differ in different tasks.

Name	Notation	Example
Input Sample	x	ప్రచారం చేయబడిన వాటిని ఖచ్చితంగా చేస్తుంది! Explanation: *absolutely does what was advertised!*
Retrieved Sample	x_r	*Great! Works as stated.*
Label	y_r	1
Prompting Pattern	$f_p(x)$	[X] *In summary, the product was* [MASK].
Verbalizer	$v(y)$	0 → 'terrible', 1 → 'great'
Prompted Input	x'	*Great! Works as stated. In summary, the product was great!* ప్రచారం చేయబడిన వాటిని ఖచ్చితంగా చేస్తుంది! *In summary, the product was* [MASK].

Table 4.1: An Example of Low-Resource Language Input Through the Pipeline

[MASK]". "[X]" is filled by the input sample. Verbalizer converts the labels into natural language answers that are predicted by the "[MASK]". The input sample integrated with retrieved information after prompting is then fed into the MPLM as prompted input for further prediction.

The difference between unlabeled and labeled pipeline lies in the way to obtain the label for retrieved sample. Figure 4.2 shows the pipeline of zero-shot on low-resource languages by cross-lingual retrieval from unlabeled high-resource language datasets. In this pipeline, the labels are obtained by self-prediction (Liu et al., 2022) using the MPLM. The retrieved sample is first predicted by the model and then prompted with the low-resource language input sample.

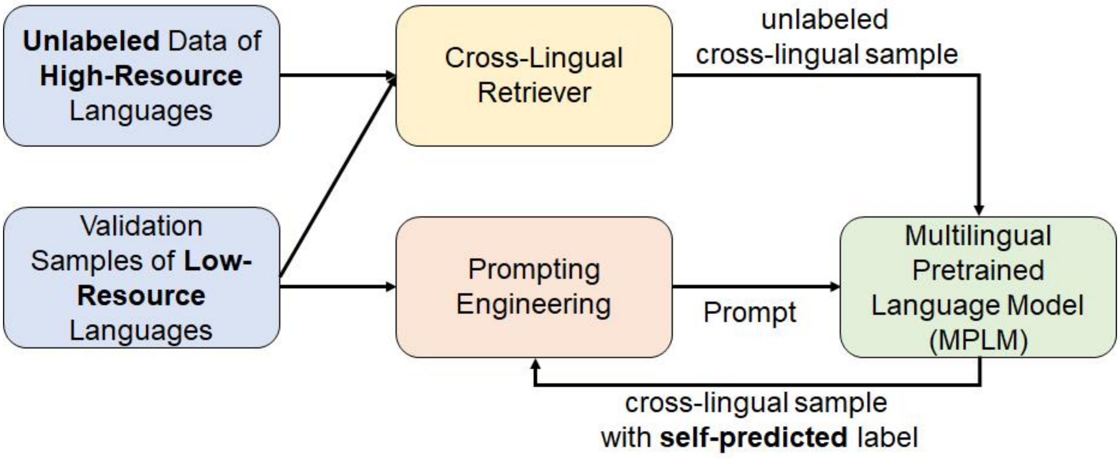

Unlabeled Data of High-Resource Languages
Validation Samples of Low-Resource Languages
Cross-Lingual Retriever
unlabeled cross-lingual sample
Prompting Engineering
Prompt
Multilingual Pretrained Language Model (MPLM)
cross-lingual sample with self-predicted label

Figure 4.2: Pipeline of zero-shot on low-resource languages by cross-lingual retrieval from **unlabeled** high-resource language datasets.

4.2 Sentence Pool of High-Resource Language

The basic idea of our proposed method is to leverage the rich language resources of high-resource languages to improve the zero-shot performance on low-resource languages. Based on this point, a sentence pool of high-resource language is established for later cross-lingual retrieval. The sentence pool can be either labeled or unlabeled. Motivated by the work by Wang et al. (2022a), who apply retrieval-based methods to the training data directly, this work uses the training sets of high-resource languages, such as English, to construct retrieval sentence pools. Under the setting of labeled high-resource language data, we extract both texts and labels from the training sets for the sentence pool; under the unlabeled setting, we only extract the text part.

4.3 Cross-Lingual Retrieval

Sparse representation based and dense representation based retrieval (see Section 2.4) are commonly used in the monolingual NLP research. For cross-lingual retrieval engine, sparse representation like BM25 model (Schütze et al., 2008) based on term frequency is not applicable, as the overlapped words across different languages are normally scarce. Dense representation based retrievers search on the basis of deep semantic representation instead of mapping between tokens on a shallow level and thus works better for cross-lingual retrieval.

In this work, we select the dense representation based method for cross-lingual retrieval. We use the multilingual sentence transformer proposed by Reimers and Gurevych (2020).

4.4 Prompting Methods

Following Schick and Schütze (2020a), we use discrete prompts (see Section 3.3) in the prompting engineering. The data samples are reformulated into cloze-style questions using a manually designed template. We not only prompt the input sample but also prompt the retrieved high-resource language sample as cross-lingual priming information. By reformulating the input text, a classification task can be converted to ask the PLM to fill in a blank in the sentence. The reformulated form is similar to the MLM task during the pretraining of the PLM. By using the verbalizer, the labels are mapped to tokens in the PLM vocabulary. The PLM fills the blank in the prompted text by comparing the probabilities of those mapped words predicted by itself and returning the word with the most probability as the answer to the blank.

To take advantage of cross-lingual information of high-resource languages, we combine not just the most semantically similar one sample from cross-lingual retrieval as the priming information with the input sample for prompting. Following (Liu et al., 2022), we use the *bag-of-contexts (BOC)* priming, which means one individual retrieved sample is used for priming and prediction each time and then compute the average of the resulting label distributions as the final prediction.

5 Experimental Settings

This chapter introduces the experimental settings. Section 5.1 introduces three tasks studied in this work and how the datasets are established. The prompt patterns used for each task are also listed in this section.Section 5.2 presents the low-resource languages used for evaluation and the high-resource languages used for retrieval. Section 5.3 presents the models used for the retrieval and zero-shot transfer and details of the experimental setup.

5.1 Datasets

We focus on validating our proposed zero-shot transfer learning methods for low-resource languages by cross-lingual retrieval with classification tasks in this work. Three representative classification tasks are selected for evaluation: binary sentiment analysis on Amazon product reviews (Keung et al., 2020), topic classification on AG News texts (Zhang et al., 2015) and natural language inference on XNLI (Conneau et al., 2018) dataset. For each dataset, we use the train set as sentence pool for cross-lingual retrieval and the test set for evaluation. Limit to the lack in annotated low-resource test datasets, we construct a multilingual parallel test datasets by machine translating annotated high-resource language datasets to low-resource languages.

Amazon Reviews Amazon product reviews dataset is used for text classification. In the original dataset, each data item is comprised of the review text, the star rating and some other meta information related to the product. Like the Amazon online shop, the dataset categorizes the reviews into 5 star ratings, from 1 to 5. To satisfy a binary classification setting, we only select the reviews with rating 1 for “negative (0)” and 5 for “positive (1)” for our experiment. The following prompt patterns are defined for an input review text x:

- $P(x) = x$ All in all, it was [MASK].

We define verbalizer v as:

- $v(0) =$ terrible , $v(1) =$ great

AG News AG is a collection of more than 1 million news articles. The AG’s news topic classification is constructed from the AG news collection. The news categories contained in the dataset are “World (0), “Sports (1), “Business (2)” and “Tech (3)”. We use the following prompt patterns for an input news text x:

- $P(x) =$ [MASK] News: x

The verbalizer that we use maps 0-3 to “World”, “Sports”, “Business” and “Tech”.

XNLI XNLI dataset is a subset of MultiNLI corpus (Williams et al., 2018). The text in each data item consists of two parts. Sentence A is the premise and sentence B is the hypothesis. NLI task is to predict what type of inference between the given premise and hypothesis. There are three task labels: entailment (0), neutral (1) and contradiction (2). We define the following patterns and verbalizers for a given sentence pair x_1 and x_2:

- $P(x_1, x_2) = x_1$? [MASK], x_2

For XNLI task, we use the following verbalizer:

- $v(0) =$ Yes , $v_A(1) =$ Maybe ,$v_A(2) =$ No

Creation of Multilingual Parallel Datasets To evaluate on as various low-resource languages as possible, we create a machine translation based multilingual parallel test datasets by leveraging the large amount of annotated high-resource language datasets. All original datasets used for creating the multilingual parallel test sets come from the Huggingface datasets (Lhoest et al., 2021). For all datasets, the data items of high-resource language are first extracted from the Huggingface datasets randomly and then machine translated into 10 low-resource languages (see section 5.2. For Amazon Review dataset, 1000 test data samples are created for each language. The test set of AG News for each language contains 2000 data items and XNLI task contains 1500 samples for each language. The test sets of all three tasks are balanced across their individual labels, which means every label has the same number of data samples in all test sets. Tabel 5.1 provides an overview of the datasets.

Task	**Dataset**	**Size (sent/lang)**	**#Label**	**Languages**
Sentiment Analysis	Amazon Reviews	1000	2	af, en, jv, mn, my, sw, ta, te, tl, ur, uz
Topic Categorization	AG News	2000	4	
Sentence Pair Classification	XNLI	1500	3	

Table 5.1: Summary of the test dataset for each task. Size refers to the number of sentences of the test data for each language.

5.2 Languages

Table 5.2 presents the list of low-resource languages evaluated in this work. As we predefined (section 2.3), languages with the Wikipedia size less the 0.177GB are regarded as low-resource languages in this study. The variety of languages was considered when deciding which low-resource languages to use. We have covered as many kinds of language family as possible. A total number of 6 languages families are included. They are Indo-European languages (Afrikaans, Urdu), Austronesian languages (Javanese, Tagalog), Altaic languages (Mongolian, Uzbek), Dravidian languages (Tamil and Telugu), Sino-Tibetan language (Burmese) and Niger-Congo language (Swahili). 10 low-resource languages are selected in total. The Wikipedia size of all languages is between 0.022GB and 0.177GB. The high-resource language English serves as the source language for cross-lingual retrieval in this work.

Lauscher et al. (2020) point out that two linguistic factors exert crucial effect on cross-lingual transfer performance: (1) size of the pretraining corpus of the target language and (2) language similarity between the source language and target language. We measure the two factors for the 10 low-resource languages used in our work for further analysis on the correlation of cross-lingual performance for our proposed method.

Pretraining Corpus Size The MPLM used in this study is mBERT. mBERT is pretrained on the multilingual Wikipedia, so the pretrainig corpus size is the same as Wikipedia size of different languages. In Table 5.2, corpus size is represented by the log_2 of the Wikipedia size in MB.

Language Similarity For each low-resource language, we calculate a index of language similarity to English and rank all languages on the basis of similarity index in an ascending order. Language similarity is used to measure the distance between two languages. Malaviya et al. (2017) and Littell et al. (2017) propose LANG2VEC, which represent languages using different types of vectors to signify the typological, geographical and phylogenetic features of languages. Based on the variety of language vectors, different types of language distance can be calculated using cosine similarity. We measure the language similarity

ISO	**Language**	**Family**	**Sim**	**Size**	**Example**
<u>en</u>	English	Indo-European	11	14	*quick shipping and works great*
af	Afrikaans	Indo-European	9	6	vinnige aflewering en werk goed
jv	Javanese	Austronesian	1	5	*Pengiriman Cepet lan Bisa Apik*
mn	Mongolian	Altaic	8	5	хурдан хүргэлт, сайн ажилладаг
my	Burmese	Sino-Tibetan	3	5	အမြန်ပို့ဆောင်ခြင်းနှင့်အကြီးအကျင့်ကိုကျင့်
sw	Swahili	Niger-Congo	5	5	*Usafirishaji wa haraka na hufanya kazi nzuri*
ta	Tamil	Dravidian	2	7	விரைவான கப்பல் மற்றும் நன்றாக வேலை செய்கிறது
te	Telugu	Dravidian	6	7	శీఘ్ర షిప్పింగ్ మరియు గొప్పగా పనిచేస్తుంది
tl	Tagalog	Austronesian	4	6	*mabilis na pagpapadala at mahusay na gumagana*
ur	Urdu	Indo-European	10	7	فوری شپنگ اور بہت اچھا کام کرتا ہے
uz	Uzbek	Altaic	7	6	*tezkor etkazib berish va ajoyib ishlaydi*

Table 5.2: Overview of low-resource languages evaluated in this work. English is the high-resource language as the source language for cross-lingual retrieval. "Sim" refers to the ascending ranking of language similarity between low-resource languages and English. "Size" refers to the language data size used for the pretraining of mBERT and is represented by the log_2 of the size in MB.

considering 5 linguistic features: syntax (SYN), phonology (PHO), phonological inventory (INV), language family (FAM) and geography (GEO). Each feature can be represented by a type of vector from LANG2VEC. SYN, PHO and INV vectors encode a linguistically typological property respectively. FAM and GEO vectors express the phylogenetic properties of languages. FAM vector denotes memberships in language families and GEO vector contains the information of orthodromic distances for languages.

	SYN		PHO		INV		FAM		GEO		**SIM**	
	SC	RNK	SC	RNK	SC	RNK	SC	RNK	SC	RNK	**SC**	**RNK**
<u>en</u>	100.0	11	100.0	11	100.0	11	100.0	11	100.0	11	**10.0**	**11**
af	52.6	10	3.0	2	69.1	3	50.5	10	86.8	2	**6.9**	**9**
jw	30.0	2	67.5	6	77.3	9	0.2	4	80.4	1	**3.5**	**1**
mn	47.2	8	90.5	8	70.7	5	0.2	5	91.4	6	**6.4**	**8**
my	37.5	3	92.0	10	86.8	10	0.1	3	87.7	3	**4.4**	**3**
sw	42.3	6	90.9	9	76.2	8	0.1	1	91.5	7	**5.2**	**5**
ta	43.3	7	90.9	1	76.2	4	0.1	2	91.5	4	**4.0**	**2**
te	38.2	4	85.3	7	74.1	7	0.2	7	89.2	5	**5.7**	**6**
tl	1.7	1	3.0	3	1.4	1	0.3	8	94.5	9	**4.9**	**2**
ur	50.0	9	3.0	4	71.6	6	12.7	9	92.5	8	**8.0**	**10**
uz	40.0	5	30.3	5	65.0	2	0.2	6	94.7	10	**6.0**	**7**

Table 5.3: Details of the similarity scores between low-resource language and English. SC: Score; RNK: Rank.

Table 5.3 displays the 5 above mentioned similarities between low-resource languages and English. For each feature, we calculate a similarity score by calculating the cosine similarity of the low-resource language vector and English language vector. At last, we compute a combined similarity score for each language by ranking each similarity in an ascending order and then weight averaging the places of the language in each ranking, as equation (5.1) shows:

$$SIM = \sum_{f \in \mathcal{F}} w_f \cdot rank_f \tag{5.1}$$

$\mathcal{F}$ is the set of features. The weights for features SYN, PHO, INV, FAM, GEO are 0.3, 0.1, 0.1, 0.3, 0.2, respectively, such that typological and phylogenetic features are balanced.

5.3 Model

We conduct all zero-shot cross-lingual experiments using the pretrained multilingual BERT "bert-base-multilingual-cased" (Devlin et al., 2018) containing 178M parameters trained on Wikipedia corpora in 104 languages. For the cross-lingual retrieval, we use the multilingual sentence transformer "paraphrase-multilingual-mpnet-base-v2" (Reimers and Gurevych, 2020) containing 278M parameters. We use PyTorch (Paszke et al., 2019) and the Huggingface (Wolf et al., 2020) framework to perform all experiments.

6 Result Analysis

This chapter will introduce the results of experiments and the analysis of the results.

6.1 Results

We compare our proposed approach with the zero-shot prompting performance without using any additional cross-lingual information (**Prompt Only**). The approach in this work contains two pipelines (see section 4.1), marked as **Labeled** and **Unlabeled** respectively.

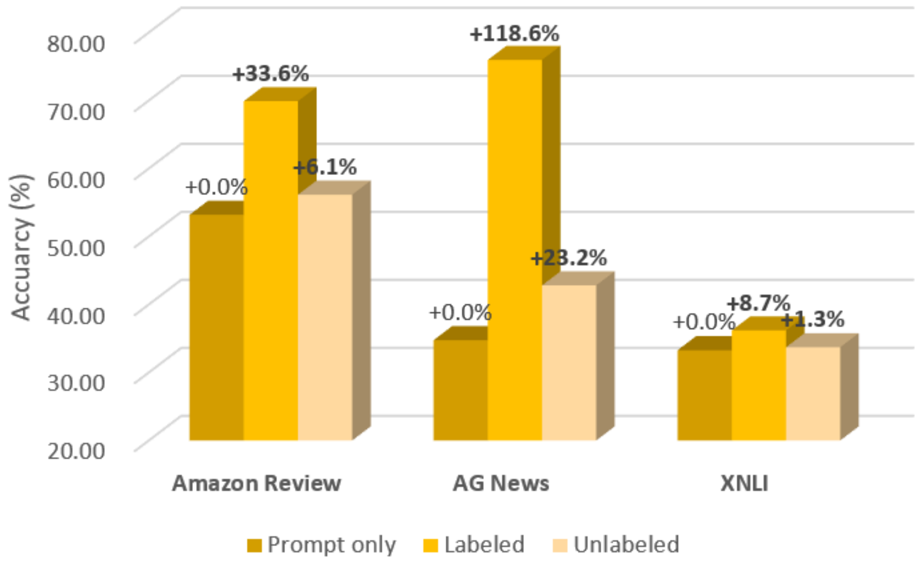


Figure 6.1: Performance improvement with cross-lingual retrieval in labeled and unlabeled settings on three tasks. Results of $k = 1$, one sample retrieved from high-resource language is used.

The performance improvement with our proposed approach can be seen in Figure 6.1. By cross-lingual retrieval, the performance has been improved for all three tasks, in both labeled and unlabeled settings. Figure 6.1 shows the results of $k = 1$, meaning that we only use one high-resource language sample. The accuracy in this figure is calculated by averaging the accuracy of 10 low-resource languages tested in this work. As can be seen, When retrieving from labeled high-resource language corpora, the performance is improved by **33.6%**, **118.6%** and **8.7%** on Amazon Review, AG News and XNLI, respectively. When retrieving from unlabeled high-resource language corpora and obtaining the label by self-prediction, the performance is improved by **6.1%**, **23.2%** and **1.3%** on three tasks respectively. The improvements on three tasks vary obviously. A large improvement happens to AG News, the topic categorization task, while XNLI performance is improved slightly. An explanation for this could be that language inference task is more difficult than topic categorization in language understanding and semantic analysis and the effect of zero-shot approach by cross-lingual retrieval depends on the difficulty of task.

In the experiment, we test the effect of different number of retrieved cross-lingual samples (k) on the zero-shot transfer performance. Table 6.1 shows the complete experimental results of binary sentiment classification task on Amazon Review dataset.

		<u>En</u>	**Af**	**Jw**	**Mn**	**My**	**Sw**	**Ta**	**Te**	**Tl**	**Ur**	**Uz**	**Avg**
MAJ		50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0
Prompt Only		58.9	50.4	54.1	51.4	56.1	51.9	56.21	53.9	51.9	54.6	51.4	53.2
$k = 1$	LB	82.4	79.2	59.3	74.9	73.8	49.9	75.4	75.8	55.4	80.6	74.9	69.9
	UN	59.1	57.9	54.2	58.6	56.0	50.5	57.2	57.5	50.9	60.0	58.6	56.2
$k = 3$	LB	86.2	81.1	61.6	79.7	77.7	54.1	79.3	78.4	57.7	84.4	79.7	73.3
	UN	57.7	56.2	53.5	56.4	55.0	50.3	55.0	55.3	50.3	57.1	56.4	54.6
$k = 5$	LB	87.2	82.9	61.5	80.6	78.9	54.4	80.5	79.0	60.2	85.0	80.6	74.3
	UN	56.0	55.0	52.9	55.3	53.6	50.0	54.0	54.0	50.1	56.4	55.3	53.7
$k = 10$	LB	88.9	85.4	**62.6**	**84.3**	81.1	**55.5**	**83.9**	81.6	**63.3**	87.3	**84.3**	**76.9**
	UN	56.0	55.8	52.5	56.3	54.2	50.0	53.9	53.5	50.3	55.5	56.3	53.8
$k = 20$	LB	**89.5**	**85.7**	61.5	83.2	**81.8**	54.4	83.1	**82.1**	62.9	**87.9**	83.2	76.6
	UN	53.6	53.5	51.7	54.5	52.8	50.0	52.9	52.9	50.4	54.0	54.5	52.7
$k = 30$	LB	88.9	85.6	61.0	83.8	81.8	54.3	83.5	82.0	62.4	87.6	83.8	76.6

Table 6.1: Results of binary sentiment classification task on Amazon Review Dataset. k is the number of retrieved cross-lingual sample. MAJ is the majority baseline. Avg is the average accuracy across 10 low-resource languages. <u>En</u> is the high-resource language for retrieval. LB: Cross-lingual retrieval from labeled high-resource data. UN: Cross-lingual retrieval from unlabeled high-resource data.

From the table we observe that: (1) Results from column Avg shows that, as expected, our proposed approach in both labeled and unlabeled settings have an positive effect on the performance. One cross-lingual retrieved sample as priming information can bring significant improvement. (2) Cross-lingual retrieval from labeled data performs much better than from self-prediction paradigm. (3) The sensitivity to cross-lingual retrieval differs from languages. Some low-resource languages, such as Swahili (Sw) and Tagalog (Tl), obtain less improvement from cross-lingual retrieval. For example, Tagalog fails to outperform prompt-only method in unlabeled setting with all numbers of retrieved sample. (4) More cross-lingual retrieved samples does not certainly bring better performance. In labeled setting, the performance is improved with the increasing of k only until $k = 10$ or 20. In unlabeled setting, the performance starts to go down after $k = 1$. With more high-resource samples, it even performs worse than the baseline.

6.2 Analysis

In this section, we will analyze the experimental results in detail to have a deeper insight into the zero-shot learning by cross-lingual retrieval.

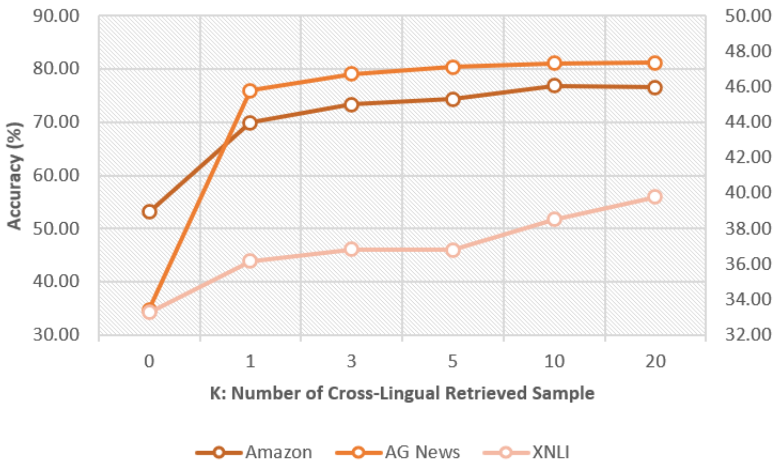


Figure 6.2: Accuracy on three tasks with different k in labeled setting.

6.2.1 Effect of k

To investigate the how the performance changes with the increasing of the number of retrieved high-resource language samples, we compare the results of the following experiments.

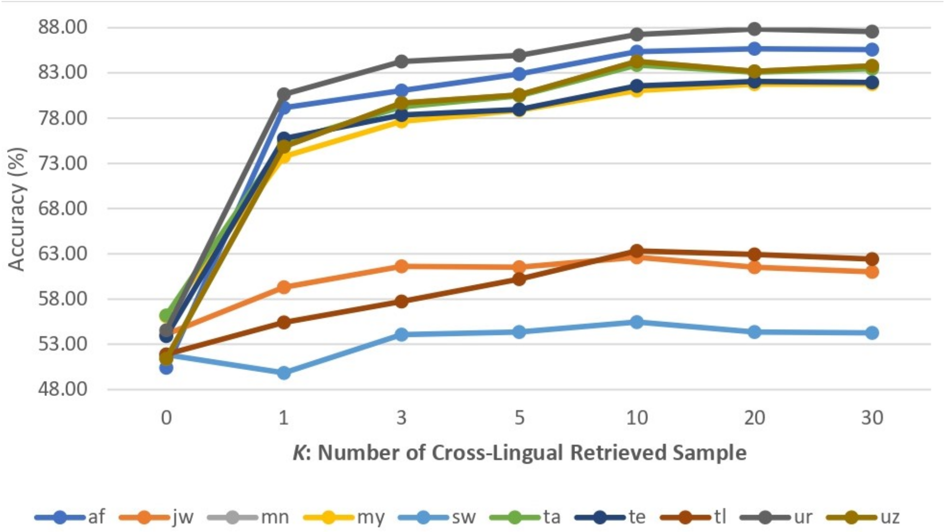


Figure 6.3: Accuracy of different low-resource languages on Amazon review task with different k in **labeled** setting.

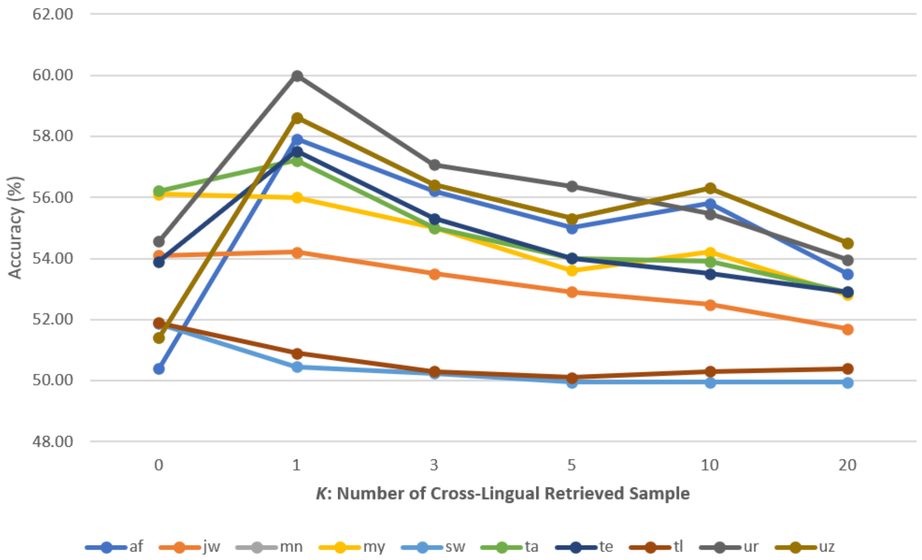


Figure 6.4: Accuracy of different low-resource languages on Amazon review task with different k in **unlabeled** setting.

We experiment with different k values on three tasks using the pipeline of cross-lingual retrieval from labeled high-resource language corpora. As Figure 6.2 shows, the accuracy increase drastically with the integration of the first cross-lingual sample. Later, the growth slows and gradually flattens in the Amazon review and AG News task. This can be explained in that the similarity of later retrieved samples and the input sample is decreasing, so the contribution to the cross-lingual transfer is getting limited. It manifests that zero-shot transfer benefits from the cross-lingual retrieval. In XNLI task, the performance does not flatten with k increasing as the previous two tasks do, and even shows a significant rise from $k = 5$ to $k = 20$. We assume that this is attributed to the property of language inference task, which detects the relationship of a sentence pair. Unlike single sentence classification task, where semantic information from similar cross-lingual sentences can be transferred directly, the transfer of sentence pair relationship learning is more complicated needs more samples.

Figure 6.3 shows the accuracy variation of different low-resource languages on Amazon review task with different k values in labeled setting. We can see the consistency of the accuracy tendency for most languages: a radical rise at the beginning, gradual slowing and flattening and even decreasing. We also analyze the accuracy variation of different low-resource languages with different k values in unlabeled setting. As is shown in Figure 6.4, for most languages, the performance is improved with the first shot and then starts to get worse.

6.2.2 Effect of Language Features

We notice that different languages have different zero-shot transfer performance with cross-lingual retrieval, so we analyze how the language features affect their transfer performance in this part. As is demonstrated in the study of Lauscher et al. (2020), language similarity and pretraining corpus size are two crucial factors for cross-lingual transfer. We use the Wikipedia size to represent the pretraining corpus size of each language, since mBERT is pretrained on multilingual Wikipedia corpora. For language similarity, we calculate a combined similarity score between each low-resource language and English following the method in section 5.2. We then rank the low-resource languages according to their pretrained corpus size and language similarity score.

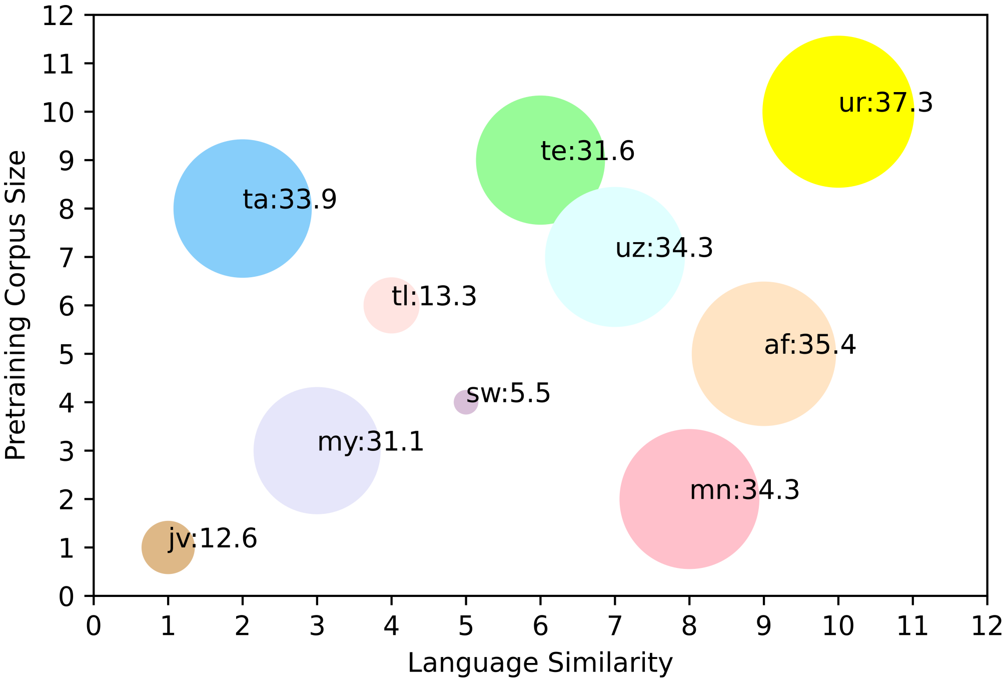


Figure 6.5: Correlation of performance of different low-resource languages with language features. The size of circles represent the accuracy of languages on Amazon review dataset with $k = 10$ in labeled setting. Label attached to each circle is the difference between accuracy and 50.00. Language similarity refers to the similarity with English. Pretraining corpus size refers to the Wikipedia size. Languages are ranked according to language similarity and pretraining corpus size in ascending sequence.

Figure 6.5 shows the correlation of different low-resource languages' performance with the language features. This figure provides an intuitive tendency that both language similarity and pretraining corpus size positively correlate to the zero-shot performance. For most languages except Burmese (My), a decent performance depends on a good ranking either in language similarity or in pretraining corpus size (such as Tamil and Mongolian). A high position in both ranking ensures an exceptional performance (such as Urdu).

6.3 Discussions

The main findings from the experiments are summarized as follows:

- We have proved that zero-shot transfer learning performance on low-resource languages can be improved by cross-lingual retrieval from both labeled and unlabeled high-resource language corpora. The improvement with labeled corpora is better than that with unlabeled corpora (+33.6% vs. +6.1% on Amazon review and +118.6% vs. +23.2%). The improvement in different tasks varies. For some tasks, this approach is less effective (118.6% on AG News vs. 8.7% on XNLI in labeled setting).
- The number of cross-lingual retrieved samples affect the performance. In labeled setting, Combining the best matched cross-lingual sample with input sample triggers a drastic rise in accuracy. However, continuing to increase k will slow down the increasing of performance and the performance will gradually flatten and even decline. In unlabeled setting, only the first shot brings obvious improvement. The increasing of k can even deteriorate the performance to a lower level than baseline.
- Zero-shot performance also correlates to the language properties in each low-resource language. For most low-resource languages, having at least one of high language similarity or large pretraining corpus size can bring a decent performance. Taking a high position in both ranking ensures an exceptional performance.

7 Conclusion

Research on zero-shot learning on low-resource languages in NLP is motivated by inherent data scarcity of low-resource languages. Multilingual Pretrained Language Models (MPLMs) have shown its strong multilinguality in recent empirical cross-lingual transfer studies. However, the multilinguality of MPLMs have been proved to be imbalanced. Low-resource languages benefit only limitedly from the advanced NLP technologies. On the contrary, languages with a large amount of both labeled and unlabeled resources are not fully utilized.

7.1 Summary

In our work, we have presented the potential to improve zero-shot transfer learning performance on low-resource languages by cross-lingual retrieval from high-resource languages. We have applied our proposed approach to three tasks (binary sentiment classification, topic categorization, natural language inference) with multilingual parallel test sets across 10 low-resource languages covering 6 language families. We have presented an empirical analysis of factors affecting transfer performance, such as the number of cross-lingual retrieved samples and language properties.

7.2 Future Work

The work will be further improved and completed in the future from the following aspects:

- More ablation study should be done to further validate the effectiveness of the proposed approach. Factors that should be considered include the prompt engineering, the cross-lingual retrieval methods, selection of high-resource languages etc.
  - **Prompt Engineering** It can be investigated how the designing of the patterns and selection of the verbalizers affects the experimental results.
  - **Cross-lingual Retrieval Methods** It should be considered how the retrieval quality and methods influence the performance. In this work, we use multilingual sentence transformer, a dense representation based retrieval method as cross-lingual retriever, other methods such as sparse representation based methods and alignment-motivated methods could be tested. Random retrieval should also be operated.
  - **High-resource Languages** We use English as the high-resource language for cross-lingual retrieval in this work. Other high-resource languages especially with different linguistic properties can be considered.
- More detailed statistical correlation analysis should be conducted to investigate the correlation between zero-shot performance and different language-related elements, such as the language similarity between high-resource language for retrieval and low-resource language, the pretraining data size of low-resource languages as well as high-resource languages.
- Our approach could be extensively applied to more tasks with more models. We have carried out all experiments using mBERT on classification tasks. Other models like XLM-R can also be used to validate the approach. Besides the tasks used in our

study, zero-shot learning by cross-lingual retrieval methods for other classification tasks and even generation tasks like text summarization can be further explored.

- We have used prompting methods to utilize cross-lingual retrieved information. We can also leverage the paradigm of few-shot finetuning and compare the performance of both methods in our future work.

Literaturverzeichnis

Agić, Ž., Johannsen, A., Plank, B., Alonso, H. M., Schluter, N., and Søgaard, A. (2016). Multilingual projection for parsing truly low-resource languages. *Transactions of the Association for Computational Linguistics*, 4:301–312.

Artetxe, M., Ruder, S., and Yogatama, D. (2019). On the cross-lingual transferability of monolingual representations. *arXiv preprint arXiv:1910.11856.*

Artetxe, M. and Schwenk, H. (2019). Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *Transactions of the Association for Computational Linguistics*, 7:597–610.

Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473.*

Bender, E. M., Gebru, T., McMillan-Major, A., and Shmitchell, S. (2021). On the dangers of stochastic parrots: Can language models be too big?. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pages 610–623.

Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Bowman, S. R., Angeli, G., Potts, C., and Manning, C. D. (2015). A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal. Association for Computational Linguistics.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. (2020). Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Cer, D., Yang, Y., Kong, S.-y., Hua, N., Limtiaco, N., John, R. S., Constant, N., Guajardo-Cespedes, M., Yuan, S., Tar, C., et al. (2018). Universal sentence encoder for english. In *Proceedings of the 2018 conference on empirical methods in natural language processing: system demonstrations*, pages 169–174.

Chen, D., Fisch, A., Weston, J., and Bordes, A. (2017). Reading wikipedia to answer open-domain questions. *arXiv preprint arXiv:1704.00051.*

Chidambaram, M., Yang, Y., Cer, D., Yuan, S., Sung, Y.-H., Strope, B., and Kurzweil, R. (2018). Learning cross-lingual sentence representations via a multi-task dual-encoder model. *arXiv preprint arXiv:1810.12836.*

Clark, J. H., Choi, E., Collins, M., Garrette, D., Kwiatkowski, T., Nikolaev, V., and Palomaki, J. (2020). Tydi qa: A benchmark for information-seeking question answering in typologically diverse languages. *Transactions of the Association for Computational Linguistics*, 8:454–470.

Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzmán, F., Grave, E., Ott, M., Zettlemoyer, L., and Stoyanov, V. (2019). Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116.*

Conneau, A., Kiela, D., Schwenk, H., Barrault, L., and Bordes, A. (2017). Supervised learning of universal sentence representations from natural language inference data. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 670–680, Copenhagen, Denmark. Association for Computational Linguistics.

Conneau, A., Lample, G., Rinott, R., Williams, A., Bowman, S. R., Schwenk, H., and Stoyanov, V. (2018). Xnli: Evaluating cross-lingual sentence representations. *arXiv preprint arXiv:1809.05053*.

Deshpande, A., Talukdar, P., and Narasimhan, K. (2021). When is bert multilingual? isolating crucial ingredients for cross-lingual transfer. *arXiv preprint arXiv:2110.14782*.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Dufter, P. and Schütze, H. (2020). Identifying necessary elements for bert's multilinguality. *arXiv preprint arXiv:2005.00396*.

Fan, A., Bhosale, S., Schwenk, H., Ma, Z., El-Kishky, A., Goyal, S., Baines, M., Celebi, O., Wenzek, G., Chaudhary, V., et al. (2021). Beyond english-centric multilingual machine translation. *Journal of Machine Learning Research*, 22(107):1–48.

Firth, J. R. (1957). A synopsis of linguistic theory, 1930-1955. *Studies in linguistic analysis*.

Gage, P. (1994). A new algorithm for data compression. *C Users Journal*, 12(2):23–38.

Gao, T., Fisch, A., and Chen, D. (2020). Making pre-trained language models better few-shot learners. *arXiv preprint arXiv:2012.15723*.

Haviv, A., Berant, J., and Globerson, A. (2021). Bertese: Learning to speak to BERT. *CoRR*, abs/2103.05327.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

Hotelling, H. (1992). Relations between two sets of variates. In *Breakthroughs in statistics*, pages 162–190. Springer.

Howard, J. and Ruder, S. (2018). Universal language model fine-tuning for text classification. *arXiv preprint arXiv:1801.06146*.

Hu, J., Ruder, S., Siddhant, A., Neubig, G., Firat, O., and Johnson, M. (2020). Xtreme: A massively multilingual multi-task benchmark for evaluating cross-lingual generalisation. In *International Conference on Machine Learning*, pages 4411–4421. PMLR.

Huang, L., Ma, S., Zhang, D., Wei, F., and Wang, H. (2022). Zero-shot cross-lingual transfer of prompt-based tuning with a unified multilingual prompt. *arXiv preprint arXiv:2202.11451*.

Jiang, Z., Xu, F. F., Araki, J., and Neubig, G. (2020). How can we know what language models know? *Transactions of the Association for Computational Linguistics*, 8:423–438.

Joshi, P., Santy, S., Budhiraja, A., Bali, K., and Choudhury, M. (2020). The state and fate of linguistic diversity and inclusion in the nlp world. *arXiv preprint arXiv:2004.09095*.

Jundi, I. and Lapesa, G. (2022). How to translate your samples and choose your shots? analyzing translate-train & few-shot cross-lingual transfer. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 129–150.

Jurafsky, D. and Martin, J. H. (2000). Speech and language processing.

Karpukhin, V., Oğuz, B., Min, S., Lewis, P., Wu, L., Edunov, S., Chen, D., and Yih, W.-t. (2020). Dense passage retrieval for open-domain question answering. *arXiv preprint arXiv:2004.04906*.

Keung, P., Lu, Y., Szarvas, G., and Smith, N. A. (2020). The multilingual amazon reviews corpus. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*.

Kiros, R., Zhu, Y., Salakhutdinov, R. R., Zemel, R., Urtasun, R., Torralba, A., and Fidler, S. (2015). Skip-thought vectors. *Advances in neural information processing systems*, 28.

Kojima, T., Gu, S. S., Reid, M., Matsuo, Y., and Iwasawa, Y. (2022). Large language models are zero-shot reasoners.

Lampinen, A. K., Dasgupta, I., Chan, S. C. Y., Matthewson, K., Tessler, M. H., Creswell, A., McClelland, J. L., Wang, J. X., and Hill, F. (2022). Can language models learn from explanations in context?

Lample, G. and Conneau, A. (2019). Cross-lingual language model pretraining. *arXiv preprint arXiv:1901.07291*.

Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., and Soricut, R. (2019). Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Lauscher, A., Ravishankar, V., Vulić, I., and Glavaš, G. (2020). From zero to hero: On the limitations of zero-shot cross-lingual transfer with multilingual transformers. *arXiv preprint arXiv:2005.00633*.

Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., and Zettlemoyer, L. (2019a). Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Lewis, P., Oğuz, B., Rinott, R., Riedel, S., and Schwenk, H. (2019b). Mlqa: Evaluating cross-lingual extractive question answering. *arXiv preprint arXiv:1910.07475*.

Lhoest, Q., Villanova del Moral, A., Jernite, Y., Thakur, A., von Platen, P., Patil, S., Chaumond, J., Drame, M., Plu, J., Tunstall, L., Davison, J., Šaško, M., Chhablani, G., Malik, B., Brandeis, S., Le Scao, T., Sanh, V., Xu, C., Patry, N., McMillan-Major, A., Schmid, P., Gugger, S., Delangue, C., Matussière, T., Debut, L., Bekman, S., Cistac, P., Goehringer, T., Mustar, V., Lagunas, F., Rush, A., and Wolf, T. (2021). Datasets: A community library for natural language processing. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 175–184, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Li, X. L. and Liang, P. (2021). Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4582–4597, Online. Association for Computational Linguistics.

Littell, P., Mortensen, D. R., Lin, K., Kairis, K., Turner, C., and Levin, L. (2017). Uriel and lang2vec: Representing languages as typological, geographical, and phylogenetic vectors. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, volume 2, pages 8–14.

Liu, P., Yuan, W., Fu, J., Jiang, Z., Hayashi, H., and Neubig, G. (2021). Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *arXiv preprint arXiv:2107.13586*.

Liu, Y., Gu, J., Goyal, N., Li, X., Edunov, S., Ghazvininejad, M., Lewis, M., and Zettlemoyer, L. (2020). Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Liu, Y., Schick, T., and Schütze, H. (2022). Semantic-oriented unlabeled priming for large-scale language models. *arXiv preprint arXiv:2202.06133*.

Malaviya, C., Neubig, G., and Littell, P. (2017). Learning language representations for typology prediction. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Copenhagen, Denmark.

May, C., Wang, A., Bordia, S., Bowman, S. R., and Rudinger, R. (2019). On measuring social biases in sentence encoders. *ArXiv*, abs/1903.10561.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Morcos, A., Raghu, M., and Bengio, S. (2018). Insights on representational similarity in neural networks with canonical correlation. *Advances in Neural Information Processing Systems*, 31.

Pan, S. J. and Yang, Q. (2009). A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. (2019). Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32.

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

Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L. (2018). Deep contextualized word representations. In *NAACL*.

Pfeiffer, J., Vulić, I., Gurevych, I., and Ruder, S. (2020). Mad-x: An adapter-based framework for multi-task cross-lingual transfer. *arXiv preprint arXiv:2005.00052*.

Pires, T., Schlinger, E., and Garrette, D. (2019). How multilingual is multilingual bert? *arXiv preprint arXiv:1906.01502*.

Qin, G. and Eisner, J. (2021). Learning how to ask: Querying lms with mixtures of soft prompts.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al. (2019). Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P. J., et al. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(140):1–67.

Reimers, N. and Gurevych, I. (2019). Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Reimers, N. and Gurevych, I. (2020). Making monolingual sentence embeddings multilingual using knowledge distillation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Ruder, S. (2019). *Neural Transfer Learning for Natural Language Processing*. PhD thesis, National University of Ireland, Galway.

Ruder, S., Vulić, I., and Søgaard, A. (2019). A survey of cross-lingual word embedding models. *Journal of Artificial Intelligence Research*, 65:569–631.

Sanh, V., Debut, L., Chaumond, J., and Wolf, T. (2019). Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Schick, T. and Schütze, H. (2020a). Exploiting cloze questions for few shot text classification and natural language inference. *arXiv preprint arXiv:2001.07676*.

Schick, T. and Schütze, H. (2020b). Few-shot text generation with pattern-exploiting training. *arXiv preprint arXiv:2012.11926*.

Schick, T. and Schütze, H. (2020c). It's not just size that matters: Small language models are also few-shot learners. *arXiv preprint arXiv:2009.07118*.

Schuster, M. and Nakajima, K. (2012). Japanese and korean voice search. In *2012 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 5149–5152. IEEE.

Schütze, H., Manning, C. D., and Raghavan, P. (2008). *Introduction to information retrieval*, volume 39. Cambridge University Press Cambridge.

Sennrich, R., Haddow, B., and Birch, A. (2015). Neural machine translation of rare words with subword units. *arXiv preprint arXiv:1508.07909*.

Shin, T., Razeghi, Y., Logan IV, R. L., Wallace, E., and Singh, S. (2020). Autoprompt: Eliciting knowledge from language models with automatically generated prompts. *arXiv preprint arXiv:2010.15980*.

Singh, A. K. (2008). Natural language processing for less privileged languages: Where do we come from? where are we going? In *Proceedings of the IJCNLP-08 Workshop on NLP for Less Privileged Languages*.

Singh, J., McCann, B., Socher, R., and Xiong, C. (2019). BERT is not an interlingua and the bias of tokenization. In *Proceedings of the 2nd Workshop on Deep Learning Approaches for Low-Resource NLP (DeepLo 2019)*, pages 47–55, Hong Kong, China. Association for Computational Linguistics.

Sokal, R. R. (1958). A statistical method for evaluating systematic relationships. *Univ. Kansas, Sci. Bull.*, 38:1409–1438.

Song, K., Tan, X., Qin, T., Lu, J., and Liu, T.-Y. (2019). Mass: Masked sequence to sequence pre-training for language generation. *arXiv preprint arXiv:1905.02450*.

Tsvetkov, Y. (2017). Opportunities and challenges in working with low-resource languages. In *Carnegie Mellon Univ., Language Technologies Institute*.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.

Wang, S., Xu, Y., Fang, Y., Liu, Y., Sun, S., Xu, R., Zhu, C., and Zeng, M. (2022a). Training data is more valuable than you think: A simple and effective method by retrieving from training data. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3170–3179, Dublin, Ireland. Association for Computational Linguistics.

Wang, X., Ruder, S., and Neubig, G. (2022b). Expanding pretrained models to thousands more languages via lexicon-based adaptation. *arXiv preprint arXiv:2203.09435.*

Wang, Z., Mayhew, S., Roth, D., et al. (2019). Cross-lingual ability of multilingual bert: An empirical study. *arXiv preprint arXiv:1912.07840.*

Webson, A. and Pavlick, E. (2022). Do prompt-based models really understand the meaning of their prompts? In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2300–2344, Seattle, United States. Association for Computational Linguistics.

Williams, A., Nangia, N., and Bowman, S. (2018). A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122. Association for Computational Linguistics.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., et al. (2020). Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations*, pages 38–45.

Wu, S., Conneau, A., Li, H., Zettlemoyer, L., and Stoyanov, V. (2019). Emerging cross-lingual structure in pretrained language models. *arXiv preprint arXiv:1911.01464.*

Wu, S. and Dredze, M. (2019). Beto, bentz, becas: The surprising cross-lingual effectiveness of bert. *arXiv preprint arXiv:1904.09077.*

Wu, S. and Dredze, M. (2020). Are all languages created equal in multilingual bert? *arXiv preprint arXiv:2005.09093.*

Yang, Y., Zhang, Y., Tar, C., and Baldridge, J. (2019a). Paws-x: A cross-lingual adversarial dataset for paraphrase identification. *arXiv preprint arXiv:1908.11828.*

Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R. R., and Le, Q. V. (2019b). Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32.

Yu, C. T. and Salton, G. (1976). Precision weighting—an effective automatic indexing method. *Journal of the ACM (JACM)*, 23(1):76–88.

Yuan, W., Neubig, G., and Liu, P. (2021). Bartscore: Evaluating generated text as text generation. In Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P., and Vaughan, J. W., editors, *Advances in Neural Information Processing Systems*, volume 34, pages 27263–27277. Curran Associates, Inc.

Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., and Artzi, Y. (2020). Bertscore: Evaluating text generation with bert. *ArXiv*, abs/1904.09675.

Zhang, X., Zhao, J. J., and LeCun, Y. (2015). Character-level convolutional networks for text classification. *ArXiv*, abs/1509.01626.

Zhang, Z., Han, X., Liu, Z., Jiang, X., Sun, M., and Liu, Q. (2019). Ernie: Enhanced language representation with informative entities. *arXiv preprint arXiv:1905.07129.*

Zhao, M. and Schütze, H. (2021). Discrete and soft prompting for multilingual models. *arXiv preprint arXiv:2109.03630.*

Zhong, Z., Friedman, D., and Chen, D. (2021). Factual probing is [MASK]: Learning vs. learning to recall. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5017–5033, Online. Association for Computational Linguistics.

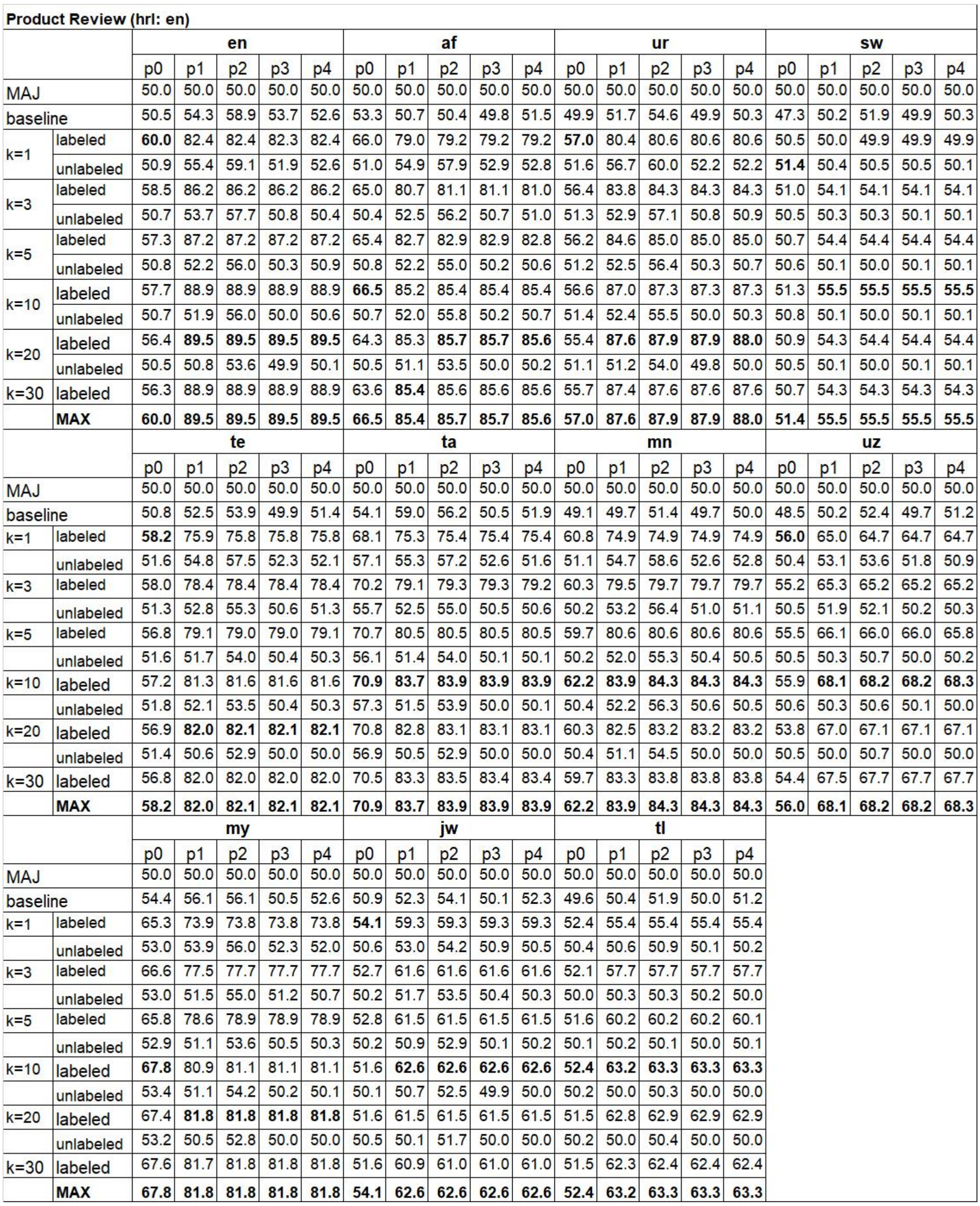
List of Figures

List of Tables

Appendix

A. Results for each task

We show the detailed experimental results for all tasks in Tables 7.1 (Amazon reviews), 7.2 (AG News), 7.3 (XNLI).

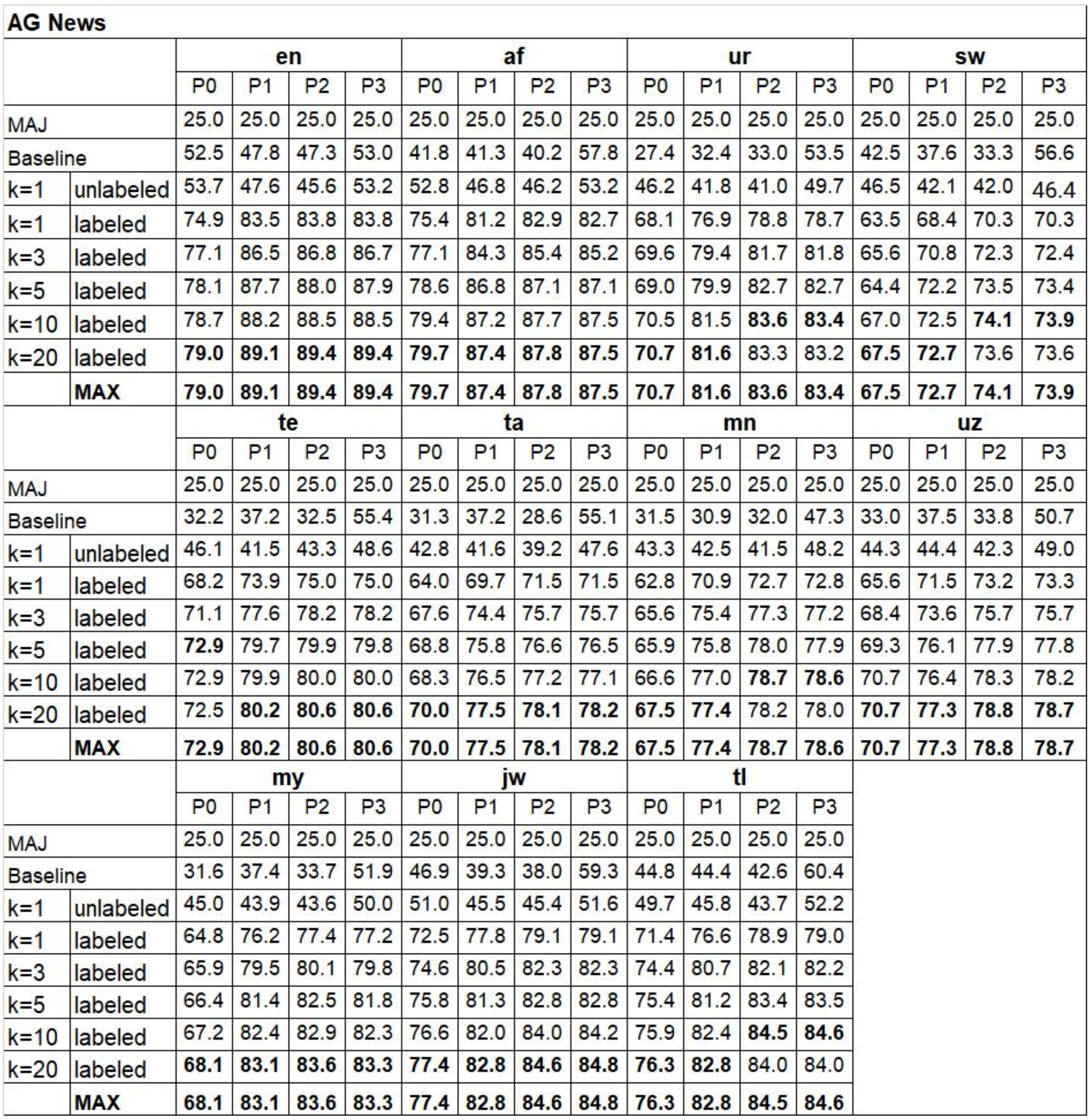


Product Review (hrl: en)																					
		en					**af**					**ur**					**sw**				
		p0	p1	p2	p3	p4	p0	p1	p2	p3	p4	p0	p1	p2	p3	p4	p0	p1	p2	p3	p4
MAJ		50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0
baseline		50.5	54.3	58.9	53.7	52.6	53.3	50.7	50.4	49.8	51.5	49.9	51.7	54.6	49.9	50.3	47.3	50.2	51.9	49.9	50.3
k=1	labeled	**60.0**	82.4	82.4	82.3	82.4	66.0	79.0	79.2	79.2	79.2	**57.0**	80.4	80.6	80.6	80.6	50.5	50.0	49.9	49.9	49.9
	unlabeled	50.9	55.4	59.1	51.9	52.6	51.0	54.9	57.9	52.9	52.8	51.6	56.7	60.0	52.2	52.2	**51.4**	50.4	50.5	50.5	50.1
k=3	labeled	58.5	86.2	86.2	86.2	86.2	65.0	80.7	81.1	81.1	81.0	56.4	83.8	84.3	84.3	84.3	51.0	54.1	54.1	54.1	54.1
	unlabeled	50.7	53.7	57.7	50.8	50.4	50.4	52.5	56.2	50.7	51.0	51.3	52.9	57.1	50.8	50.9	50.5	50.3	50.3	50.1	50.1
k=5	labeled	57.3	87.2	87.2	87.2	87.2	65.4	82.7	82.9	82.9	82.8	56.2	84.6	85.0	85.0	85.0	50.7	54.4	54.4	54.4	54.4
	unlabeled	50.8	52.2	56.0	50.3	50.9	50.8	52.2	55.0	50.2	50.6	51.2	52.5	56.4	50.3	50.7	50.6	50.1	50.0	50.1	50.1
k=10	labeled	57.7	88.9	88.9	88.9	88.9	**66.5**	85.2	85.4	85.4	85.4	56.6	87.0	87.3	87.3	87.3	51.3	**55.5**	**55.5**	**55.5**	**55.5**
	unlabeled	50.7	51.9	56.0	50.0	50.6	50.7	52.0	55.8	50.2	50.7	51.4	52.4	55.5	50.0	50.3	50.8	50.1	50.0	50.1	50.1
k=20	labeled	56.4	**89.5**	**89.5**	**89.5**	**89.5**	64.3	85.3	**85.7**	**85.7**	**85.6**	55.4	**87.6**	**87.9**	**87.9**	**88.0**	50.9	54.3	54.4	54.4	54.4
	unlabeled	50.5	50.8	53.6	49.9	50.1	50.5	51.1	53.5	50.0	50.2	51.1	51.2	54.0	49.8	50.0	50.5	50.1	50.0	50.1	50.1
k=30	labeled	56.3	88.9	88.9	88.9	88.9	63.6	**85.4**	85.6	85.6	85.6	55.7	87.4	87.6	87.6	87.6	50.7	54.3	54.3	54.3	54.3
	MAX	**60.0**	**89.5**	**89.5**	**89.5**	**89.5**	**66.5**	**85.4**	**85.7**	**85.7**	**85.6**	**57.0**	**87.6**	**87.9**	**87.9**	**88.0**	**51.4**	**55.5**	**55.5**	**55.5**	**55.5**

		te					**ta**					**mn**					**uz**				
		p0	p1	p2	p3	p4	p0	p1	p2	p3	p4	p0	p1	p2	p3	p4	p0	p1	p2	p3	p4
MAJ		50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0
baseline		50.8	52.5	53.9	49.9	51.4	54.1	59.0	56.2	50.5	51.9	49.1	49.7	51.4	49.7	50.0	48.5	50.2	52.4	49.7	51.2
k=1	labeled	**58.2**	75.9	75.8	75.8	75.8	68.1	75.3	75.4	75.4	75.4	60.8	74.9	74.9	74.9	74.9	**56.0**	65.0	64.7	64.7	64.7
	unlabeled	51.6	54.8	57.5	52.3	52.1	57.1	55.3	57.2	52.6	51.6	51.1	54.7	58.6	52.6	52.8	50.4	53.1	53.6	51.8	50.9
k=3	labeled	58.0	78.4	78.4	78.4	78.4	70.2	79.1	79.3	79.3	79.2	60.3	79.5	79.7	79.7	79.7	55.2	65.3	65.2	65.2	65.2
	unlabeled	51.3	52.8	55.3	50.6	51.3	55.7	52.5	55.0	50.5	50.6	50.2	53.2	56.4	51.0	51.1	50.5	51.9	52.1	50.2	50.3
k=5	labeled	56.8	79.1	79.0	79.0	79.1	70.7	80.5	80.5	80.5	80.5	59.7	80.6	80.6	80.6	80.6	55.5	66.1	66.0	66.0	65.8
	unlabeled	51.6	51.7	54.0	50.4	50.3	56.1	51.4	54.0	50.1	50.1	50.2	52.0	55.3	50.4	50.5	50.5	50.3	50.7	50.0	50.2
k=10	labeled	57.2	81.3	81.6	81.6	81.6	**70.9**	**83.7**	**83.9**	**83.9**	**83.9**	**62.2**	**83.9**	**84.3**	**84.3**	**84.3**	55.9	**68.1**	**68.2**	**68.2**	**68.3**
	unlabeled	51.8	52.1	53.5	50.4	50.3	57.3	51.5	53.9	50.0	50.1	50.4	52.2	56.3	50.6	50.5	50.6	50.3	50.6	50.1	50.0
k=20	labeled	56.9	**82.0**	**82.1**	**82.1**	**82.1**	70.8	82.8	83.1	83.1	83.1	60.3	82.5	83.2	83.2	83.2	53.8	67.0	67.1	67.1	67.1
	unlabeled	51.4	50.6	52.9	50.0	50.0	56.9	50.5	52.9	50.0	50.0	50.4	51.1	54.5	50.0	50.0	50.5	50.0	50.7	50.0	50.0
k=30	labeled	56.8	82.0	82.0	82.0	82.0	70.5	83.3	83.5	83.4	83.4	59.7	83.3	83.8	83.8	83.8	54.4	67.5	67.7	67.7	67.7
	MAX	**58.2**	**82.0**	**82.1**	**82.1**	**82.1**	**70.9**	**83.7**	**83.9**	**83.9**	**83.9**	**62.2**	**83.9**	**84.3**	**84.3**	**84.3**	**56.0**	**68.1**	**68.2**	**68.2**	**68.3**

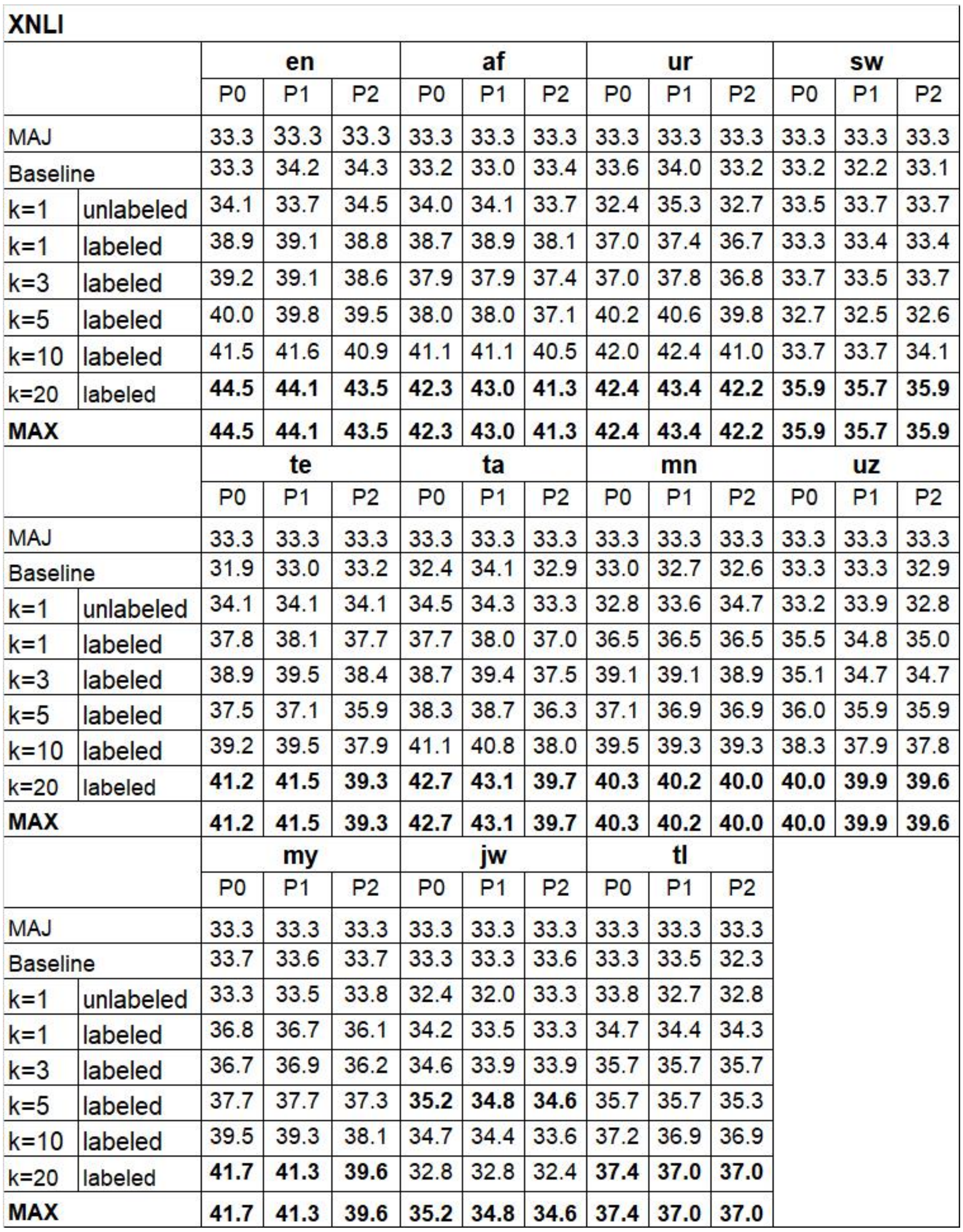
		my					**jw**					**tl**				
		p0	p1	p2	p3	p4	p0	p1	p2	p3	p4	p0	p1	p2	p3	p4
MAJ		50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0
baseline		54.4	56.1	56.1	50.5	52.6	50.9	52.3	54.1	50.1	52.3	49.6	50.4	51.9	50.0	51.2
k=1	labeled	65.3	73.9	73.8	73.8	73.8	**54.1**	59.3	59.3	59.3	59.3	52.4	55.4	55.4	55.4	55.4
	unlabeled	53.0	53.9	56.0	52.3	52.0	50.6	53.0	54.2	50.9	50.5	50.4	50.6	50.9	50.1	50.2
k=3	labeled	66.6	77.5	77.7	77.7	77.7	52.7	61.6	61.6	61.6	61.6	52.1	57.7	57.7	57.7	57.7
	unlabeled	53.0	51.5	55.0	51.2	50.7	50.2	51.7	53.5	50.4	50.3	50.0	50.3	50.3	50.2	50.0
k=5	labeled	65.8	78.6	78.9	78.9	78.9	52.8	61.5	61.5	61.5	61.5	51.6	60.2	60.2	60.2	60.1
	unlabeled	52.9	51.1	53.6	50.5	50.3	50.2	50.9	52.9	50.1	50.2	50.1	50.2	50.1	50.0	50.1
k=10	labeled	**67.8**	80.9	81.1	81.1	81.1	51.6	**62.6**	**62.6**	**62.6**	**62.6**	**52.4**	**63.2**	**63.3**	**63.3**	**63.3**
	unlabeled	53.4	51.1	54.2	50.2	50.1	50.1	50.7	52.5	49.9	50.0	50.2	50.0	50.3	50.0	50.0
k=20	labeled	67.4	**81.8**	**81.8**	**81.8**	**81.8**	51.6	61.5	61.5	61.5	61.5	51.5	62.8	62.9	62.9	62.9
	unlabeled	53.2	50.5	52.8	50.0	50.0	50.5	50.1	51.7	50.0	50.0	50.2	50.0	50.4	50.0	50.0
k=30	labeled	67.6	81.7	81.8	81.8	81.8	51.6	60.9	61.0	61.0	61.0	51.5	62.3	62.4	62.4	62.4
	MAX	**67.8**	**81.8**	**81.8**	**81.8**	**81.8**	**54.1**	**62.6**	**62.6**	**62.6**	**62.6**	**52.4**	**63.2**	**63.3**	**63.3**	**63.3**

Table 7.1: Results on Amazon reviews dataset.



AG News																	
		en				**af**				**ur**				**sw**			
		P0	P1	P2	P3	P0	P1	P2	P3	P0	P1	P2	P3	P0	P1	P2	P3
MAJ		25.0	25.0	25.0	25.0	25.0	25.0	25.0	25.0	25.0	25.0	25.0	25.0	25.0	25.0	25.0	25.0
Baseline		52.5	47.8	47.3	53.0	41.8	41.3	40.2	57.8	27.4	32.4	33.0	53.5	42.5	37.6	33.3	56.6
k=1	unlabeled	53.7	47.6	45.6	53.2	52.8	46.8	46.2	53.2	46.2	41.8	41.0	49.7	46.5	42.1	42.0	46.4
k=1	labeled	74.9	83.5	83.8	83.8	75.4	81.2	82.9	82.7	68.1	76.9	78.8	78.7	63.5	68.4	70.3	70.3
k=3	labeled	77.1	86.5	86.8	86.7	77.1	84.3	85.4	85.2	69.6	79.4	81.7	81.8	65.6	70.8	72.3	72.4
k=5	labeled	78.1	87.7	88.0	87.9	78.6	86.8	87.1	87.1	69.0	79.9	82.7	82.7	64.4	72.2	73.5	73.4
k=10	labeled	78.7	88.2	88.5	88.5	79.4	87.2	87.7	87.5	70.5	81.5	**83.6**	**83.4**	67.0	72.5	**74.1**	**73.9**
k=20	labeled	**79.0**	**89.1**	**89.4**	**89.4**	**79.7**	**87.4**	**87.8**	**87.5**	**70.7**	**81.6**	83.3	83.2	**67.5**	**72.7**	73.6	73.6
	MAX	**79.0**	**89.1**	**89.4**	**89.4**	**79.7**	**87.4**	**87.8**	**87.5**	**70.7**	**81.6**	**83.6**	**83.4**	**67.5**	**72.7**	**74.1**	**73.9**
		te				**ta**				**mn**				**uz**			
		P0	P1	P2	P3	P0	P1	P2	P3	P0	P1	P2	P3	P0	P1	P2	P3
MAJ		25.0	25.0	25.0	25.0	25.0	25.0	25.0	25.0	25.0	25.0	25.0	25.0	25.0	25.0	25.0	25.0
Baseline		32.2	37.2	32.5	55.4	31.3	37.2	28.6	55.1	31.5	30.9	32.0	47.3	33.0	37.5	33.8	50.7
k=1	unlabeled	46.1	41.5	43.3	48.6	42.8	41.6	39.2	47.6	43.3	42.5	41.5	48.2	44.3	44.4	42.3	49.0
k=1	labeled	68.2	73.9	75.0	75.0	64.0	69.7	71.5	71.5	62.8	70.9	72.7	72.8	65.6	71.5	73.2	73.3
k=3	labeled	71.1	77.6	78.2	78.2	67.6	74.4	75.7	75.7	65.6	75.4	77.3	77.2	68.4	73.6	75.7	75.7
k=5	labeled	**72.9**	79.7	79.9	79.8	68.8	75.8	76.6	76.5	65.9	75.8	78.0	77.9	69.3	76.1	77.9	77.8
k=10	labeled	72.9	79.9	80.0	80.0	68.3	76.5	77.2	77.1	66.6	77.0	**78.7**	**78.6**	70.7	76.4	78.3	78.2
k=20	labeled	72.5	**80.2**	**80.6**	**80.6**	**70.0**	**77.5**	**78.1**	**78.2**	**67.5**	**77.4**	78.2	78.0	**70.7**	**77.3**	**78.8**	**78.7**
	MAX	**72.9**	**80.2**	**80.6**	**80.6**	**70.0**	**77.5**	**78.1**	**78.2**	**67.5**	**77.4**	**78.7**	**78.6**	**70.7**	**77.3**	**78.8**	**78.7**
		my				**jw**				**tl**							
		P0	P1	P2	P3	P0	P1	P2	P3	P0	P1	P2	P3				
MAJ		25.0	25.0	25.0	25.0	25.0	25.0	25.0	25.0	25.0	25.0	25.0	25.0				
Baseline		31.6	37.4	33.7	51.9	46.9	39.3	38.0	59.3	44.8	44.4	42.6	60.4				
k=1	unlabeled	45.0	43.9	43.6	50.0	51.0	45.5	45.4	51.6	49.7	45.8	43.7	52.2				
k=1	labeled	64.8	76.2	77.4	77.2	72.5	77.8	79.1	79.1	71.4	76.6	78.9	79.0				
k=3	labeled	65.9	79.5	80.1	79.8	74.6	80.5	82.3	82.3	74.4	80.7	82.1	82.2				
k=5	labeled	66.4	81.4	82.5	81.8	75.8	81.3	82.8	82.8	75.4	81.2	83.4	83.5				
k=10	labeled	67.2	82.4	82.9	82.3	76.6	82.0	84.0	84.2	75.9	82.4	**84.5**	**84.6**				
k=20	labeled	**68.1**	**83.1**	**83.6**	**83.3**	**77.4**	**82.8**	**84.6**	**84.8**	**76.3**	**82.8**	84.0	84.0				
	MAX	**68.1**	**83.1**	**83.6**	**83.3**	**77.4**	**82.8**	**84.6**	**84.8**	**76.3**	**82.8**	**84.5**	**84.6**				

Table 7.2: Results on AG News dataset.



XNLI													
		en			**af**			**ur**			**sw**		
		P0	P1	P2	P0	P1	P2	P0	P1	P2	P0	P1	P2
MAJ		33.3	33.3	33.3	33.3	33.3	33.3	33.3	33.3	33.3	33.3	33.3	33.3
Baseline		33.3	34.2	34.3	33.2	33.0	33.4	33.6	34.0	33.2	33.2	32.2	33.1
k=1	unlabeled	34.1	33.7	34.5	34.0	34.1	33.7	32.4	35.3	32.7	33.5	33.7	33.7
k=1	labeled	38.9	39.1	38.8	38.7	38.9	38.1	37.0	37.4	36.7	33.3	33.4	33.4
k=3	labeled	39.2	39.1	38.6	37.9	37.9	37.4	37.0	37.8	36.8	33.7	33.5	33.7
k=5	labeled	40.0	39.8	39.5	38.0	38.0	37.1	40.2	40.6	39.8	32.7	32.5	32.6
k=10	labeled	41.5	41.6	40.9	41.1	41.1	40.5	42.0	42.4	41.0	33.7	33.7	34.1
k=20	labeled	**44.5**	**44.1**	**43.5**	**42.3**	**43.0**	**41.3**	**42.4**	**43.4**	**42.2**	**35.9**	**35.7**	**35.9**
MAX		**44.5**	**44.1**	**43.5**	**42.3**	**43.0**	**41.3**	**42.4**	**43.4**	**42.2**	**35.9**	**35.7**	**35.9**
		te			**ta**			**mn**			**uz**		
		P0	P1	P2	P0	P1	P2	P0	P1	P2	P0	P1	P2
MAJ		33.3	33.3	33.3	33.3	33.3	33.3	33.3	33.3	33.3	33.3	33.3	33.3
Baseline		31.9	33.0	33.2	32.4	34.1	32.9	33.0	32.7	32.6	33.3	33.3	32.9
k=1	unlabeled	34.1	34.1	34.1	34.5	34.3	33.3	32.8	33.6	34.7	33.2	33.9	32.8
k=1	labeled	37.8	38.1	37.7	37.7	38.0	37.0	36.5	36.5	36.5	35.5	34.8	35.0
k=3	labeled	38.9	39.5	38.4	38.7	39.4	37.5	39.1	39.1	38.9	35.1	34.7	34.7
k=5	labeled	37.5	37.1	35.9	38.3	38.7	36.3	37.1	36.9	36.9	36.0	35.9	35.9
k=10	labeled	39.2	39.5	37.9	41.1	40.8	38.0	39.5	39.3	39.3	38.3	37.9	37.8
k=20	labeled	**41.2**	**41.5**	**39.3**	**42.7**	**43.1**	**39.7**	**40.3**	**40.2**	**40.0**	**40.0**	**39.9**	**39.6**
MAX		**41.2**	**41.5**	**39.3**	**42.7**	**43.1**	**39.7**	**40.3**	**40.2**	**40.0**	**40.0**	**39.9**	**39.6**
		my			**jw**			**tl**					
		P0	P1	P2	P0	P1	P2	P0	P1	P2			
MAJ		33.3	33.3	33.3	33.3	33.3	33.3	33.3	33.3	33.3			
Baseline		33.7	33.6	33.7	33.3	33.3	33.6	33.3	33.5	32.3			
k=1	unlabeled	33.3	33.5	33.8	32.4	32.0	33.3	33.8	32.7	32.8			
k=1	labeled	36.8	36.7	36.1	34.2	33.5	33.3	34.7	34.4	34.3			
k=3	labeled	36.7	36.9	36.2	34.6	33.9	33.9	35.7	35.7	35.7			
k=5	labeled	37.7	37.7	37.3	**35.2**	**34.8**	**34.6**	35.7	35.7	35.3			
k=10	labeled	39.5	39.3	38.1	34.7	34.4	33.6	37.2	36.9	36.9			
k=20	labeled	**41.7**	**41.3**	**39.6**	32.8	32.8	32.4	**37.4**	**37.0**	**37.0**			
MAX		**41.7**	**41.3**	**39.6**	**35.2**	**34.8**	**34.6**	**37.4**	**37.0**	**37.0**			

Table 7.3: Results on XNLI dataset.

Inhalt der beigelegten CD

- Electronic version of the thesis in original format and in PDF format.
- Source codes created for the Master work.