UoB_UK at SemEval 2021 Task 2: Zero-Shot and Few-Shot Learning for Multi-lingual and Cross-lingual Word Sense Disambiguation.

Wei Li, Harish Tayyar Madabushi and **Mark Lee**

School of Computer Science
University of Birmingham
United Kingdom

WXL885@student.bham.ac.uk
Harish@HarishTayyarMadabushi.com, M.G.Lee@bham.ac.uk

Abstract

This paper describes our submission to SemEval 2021 Task 2. We compare XLM-RoBERTa Base and Large in the few-shot and zero-shot settings and additionally test the effectiveness of using a k-nearest neighbors classifier in the few-shot setting instead of the more traditional multi-layered perceptron. Our experiments on both the multi-lingual and cross-lingual data show that XLM-RoBERTa Large, unlike the Base version, seems to be able to more effectively transfer learning in a few-shot setting and that the k-nearest neighbors classifier is indeed a more powerful classifier than a multi-layered perceptron when used in few-shot learning.

1 Introduction and Motivation

Word Sense Disambiguation (WSD) is the task of disambiguating semantic meaning at the word level and is an important part of Natural Language Processing (NLP) with applications in several downstream tasks (Wang et al., 2020b). In recent years, Few Shot Learning (FSL) has been successful in several domains (Wang et al., 2020a) including in NLP (Yan et al., 2018). Modern deep neural networks require a significant amount of training data that might not always be available. FSL is a solution to this problem, wherein training data from a related domain or language can be used to augment training on the target domain/language with significantly less data. FSL can be characterised as n-way k-shot classification.

We participate in SemEval Task 2 Multilingual and Cross-lingual Word in Context Disambiguation (Martelli et al., 2021), which provides 8,000 training examples in English but only 32 in the other target languages. In addition the tasks requires disambiguation of word pairs in the cross lingual setting between EN-AR, EN-FR, EN-RU, and EN-ZH, where only 8 examples each are available. This sparsity in training data led us to explore the use of FSL for this task.

We hypothesise that:

1. “Large” models, which have been shown to have a lot more linguistic information and high-level generalisability, are more likely to be able to generalise their learning in the few-shot setting, and

2. that the use of a k-nearest neighbors (KNN) classifier is likely to be more effective in the few-shot setting. This hypothesis is based on our exploration of related work (Section 2).

Our experiments on both the multi-lingual and cross-lingual data show that these hypotheses are in fact correct. We show that XLM-RoBERTa Large, unlike the Base version, seems to be able to more effectively transfer knowledge in a few-shot setting, and that the KNN classifier is indeed a more powerful classifier than a multi-layered perceptron (MLP) when used in few-shot learning. To ensure reproducibility and so other researchers can build on this work, we release the program code, hyperparameters and experiments associated with this work[1].

2 Related work

The first effective method that implemented Few Shot Learning in NLP was that by Koch et al. (2015), who introduced the application of the siamese network in one-shot learning. The siamese network is typically used to calculate semantic similarity between sentences and was shown to be powerful in the FSL setting.

A recent use of zero-shot learning in WSD was work by Kumar et al. (2019), who proposed the

[1] https://github.com/weilk/SemEval-2021-Task-2

Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021), pages 738–742
Bangkok, Thailand (online), August 5–6, 2021. ©2021 Association for Computational Linguistics

extension of WSD systems by incorporating Sense embeddings (EWISE). These sense embeddings are derived from a knowledge graph, namely WordNet (Miller, 1998), and graph embeddings. EWISE predicts over an embedding space instead of the discrete label space and allow generalized zero-shot learning capability. Instead of using annotated data, it uses definitions from WordNet.

Pelevina et al. (2017) proposed a simple and effective method which uses clustering in ego-networks. Egocentric networks are local networks with one central node, known as the “ego”. This method allows labeling of words in the context with learned sense vectors thus providing a new approach to unsupervised WSD.

Our use of the K Nearest Neighbours (KNN) classifier is motivated by the work by Snell et al. (2017), who proposed a very straightforward network, similar to the nearest class mean approach (Mensink et al., 2013), that makes use of the classes of “prototypical” examples to classify new examples based on their distances. We extend this method by use of pre-trained models and a KNN classifier (Section 3).

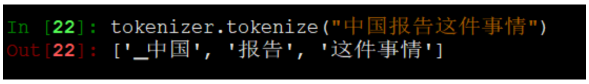
3 Methodology

In this section, we describe the different models used for the task. Our models are built on top of XLM-RoBERTa, but because cross-lingual data is limited, we used FSL during training. We also perform data preprocessing, especially in the case of Chinese and Arabic where tokenisation is inexact.

3.1 Data Preprocessing

We only use data available from the official data set provided[2] for all our experiments. We performed additional preprocessing in the case of Chinese and Arabic as it is often difficult to locate the target word in these cases. This is because Conneau et al. (2020) use the SentencePiece algorithm as the basis of XLMRobertaTokenizer, which tends to output the largest granularity words by meaning in both Chinese and Arabic. Due to this, it is possible that the target word may either be included in the hypernym’s word-piece or be cut and included in a different hypernym’s word-piece.

For example, XLM-RoBERTa tokeniser was used to tokenise a Chinese sentence, and the output is shown in Figure 1. The target word is “事情”. However, the tokeniser includes the target word in the larger word “这件事情”. To get around this and to ensure correct word tokenisation, we add a comma (“,”) around the target word in both Chinese and Arabic sentences as in: “中国报告这件,事情,”.



```
In [22]: tokenizer.tokenize("中国报告这件事情")
Out[22]: ['_中国', '报告', '这件事情']
```

Figure 1: Tokenise Chinense sentence

3.2 System Architecture

In order to find the most effective model architecture, we experiment with different variations: The cosine similarity between contextual word representations obtained from XLM-RoBERTa (Conneau et al., 2020) using multiple thresholds, classification of pre-generated XLM-RoBERTa embeddings using a multi-layer perceptron, and finally an end to end model with a softmax layer for classification. Our experiments showed the end to end model to be most effective, which we use for all downstream experimentation.

3.2.1 The Base End-to-end Model

Models used in this work are variations of the base end-to-end model detailed in Figure 2. The model takes as input the two sentences and the positions of the target words. Each of the input sentences are transformed into the contextual word embeddings using XML-RoBERTa and from the resultant embeddings the contextual embeddings associated with the target word are selected. These vectors, v_1 and v_2 are further augmented with their difference and concatenated into a single vector $v_1, v_2, v_1 - v_2, v_2 - v_1$. We note that although adding the vectors $v_1 - v_2$ and $v_2 - v_1$ does not provide the model with any additional information, our initial experiments showed that this boosted performance.

This combined vector is then passed through a transformer layer and a mean pool layer splits the output of the transformer layer into two different vectors which are compared using cosine similarity. The cosine similarity of these vectors is used to build a vector $(cos, 1 - cos)$ which is then passed through either an MLP or a KNN classifier. This output is finally passed through a softmax layer to classify the word in the two sentences as belonging to the same meaning or not.

[2] https://github.com/SapienzaNLP/mcl-wic

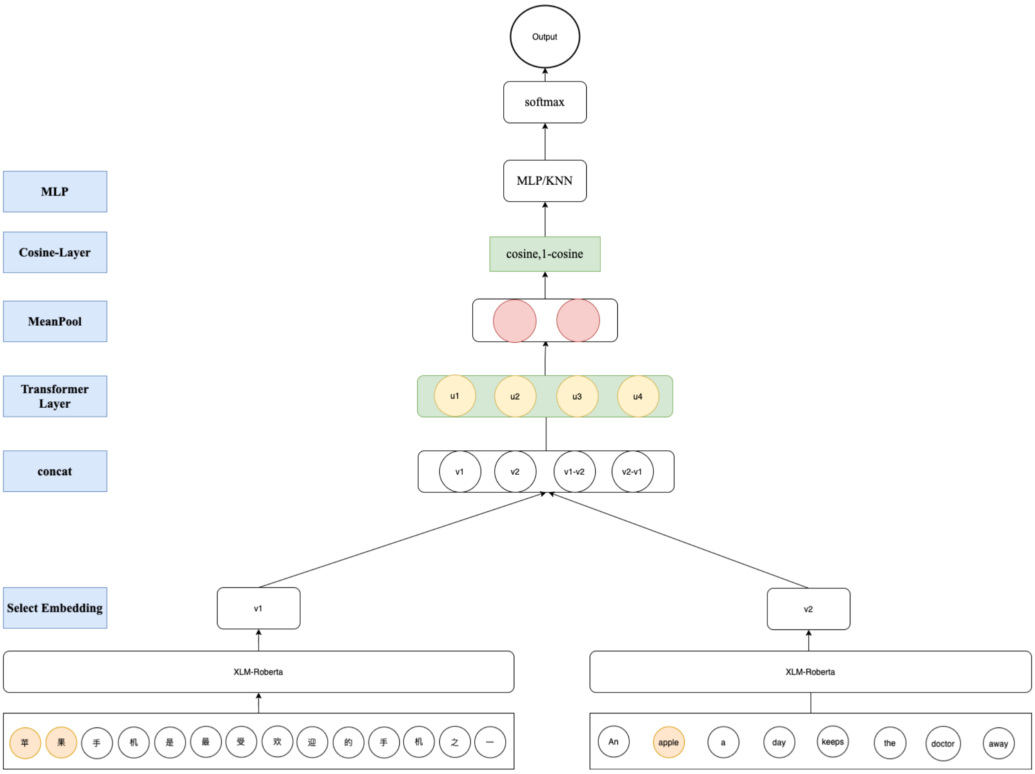


Figure 2: The architecture of the model used for sense disambiguation.

3.2.2 Zero-Shot Models

The zero-shot methods we use are variations of the base model, one which uses the Base version of XLM-RoBERTa, and the other which uses the Large version of XLM-RoBERTa. In this setting we use an MLP classifier instead of the KNN classifier. These models are trained on the English training data and tested on the multi-lingual and cross-lingual settings.

3.2.3 Few-Shot Models

The models that we use in the few-shot setting are modifications of the zero-shot model and we use the zero-shot model as a baseline. In all cases we start with the model trained on the English training data. The first variation replaces the multi-layered perceptron with a KNN classifier (with $k = 2$), and the second variation replaces XLM-RoBERTa Base with XLM-RoBERTa Large. These variations are further detailed in Section 4 and are trained on the minimal data available in the target language pairs.

4 Experiments Design

We design several experiments to find the best performing model. All the experiments are performed on the same platform[3]. We also use warmup in all the experiments and further hyperparameters are detailed in the documentation associated with the program code released with this work. For each experiment, five different random seed are tested and we choose the best performing model.

We use the English training data to train our baseline and zero-shot models. The development data provided consisted of 1000 examples for each language in the multi-lingual setting (en-en, ar-ar, zh-zh, fr-fr, and ru-ru). We divide this into a Dev-Train subset consisting of 600 examples, a Dev-Validation and Dev-Test subsets consisting of 200 examples each. In the cross-lingual setting, where we have only 8 examples from each language pair, we use the trial data for few shot training and do not use a validation or test set due to data sparsity. Finally, due to data sparsity in the target languages we freeze the transformer layers of XLM-RoBERTa during the few shot training phase.

Based on the results of these experiments, we select the best five models to submit to the official websites. These models were:

[3] System: Ubuntu 16.04.6 LTS. CPU:Intel(R) Xeon(R) CPU E5-2680 v3 @ 2.50GHz. 4 cores,8 threads. GPU:GeForce RTX 3090 24GB. Pytorch+cuda:1.7.1+cu110. Python 3.7.4

	EN-AR	EN-FR	EN-RU	EN-ZH
XLM-RoBERTa Base + MLP, Zero-Shot	74.30 (46)	80.00 (39)	81.60 (35)	76.30 (43)
XLM-RoBERTa Large + MLP, Zero-Shot	76.70 (41)	**84.00 (19)**	82.90 (28)	81.00 (37)
XLM-RoBERTa Base + MLP, Few-shot	73.00 (49)	76.50 (50)	80.10 (40)	75.50 (44)
XLM-RoBERTa Large + MLP, Few-shot	80.40 (34)	81.40 (34)	80.70 (38)	81.80 (33)
XLM-RoBERTa Large + KNN, Few-shot	**81.90 (30)**	83.90 (20)	**83.30 (24)**	**83.60 (29)**

Table 1: Accuracy on the final cross-lingual test set with the rank achieved by that submission in brackets. The highest score for each language pair is highlighted in bold. Rows 1, 3 and 2, 4 are comparable results between zero-shot and few-shot settings.

	EN-EN	AR-AR	FR-FR	RU-RU	ZH-ZH
XLM-RoBERTa Base + MLP, Zero-Shot	84.50 (50)	78.20 (40)	78.60 (44)	78.10 (34)	81.40 (32)
XLM-RoBERTa Large + MLP, Zero-Shot	87.30 (37)	77.30 (43)	**84.20 (18)**	**82.30 (23)**	80.80 (35)
XLM-RoBERTa Base + MLP, Few-shot	84.40 (51)	78.90 (36)	79.20 (41)	78.10 (34)	80.60 (36)
XLM-RoBERTa Large + MLP, Few-shot	87.10 (38)	**81.00 (27)**	83.40 (22)	82.00 (24)	82.00 (28)
XLM-RoBERTa Large + KNN, Few-shot	**88.50 (33)**	78.40 (38)	83.60 (21)	81.90 (25)	**82.10 (27)**

Table 2: Accuracy on the final multi-lingual test set with the rank achieved by that submission in brackets. The highest score for each language pair is highlighted in bold. Rows 1, 3 and 2, 4 are comparable results between Zero-Shot and few shot settings. The variation in en-en is a result of random initialisation.

1. XLM-RoBERTa Base with a multi-layered perceptron as the **baseline** zero-shot model.
2. XLM-RoBERTa Large with a multi-layered perceptron as a second zero-shot model.
3. XLM-RoBERTa Base with a multi-layered perceptron, additionally trained on available target language data as a few-shot model.
4. XLM-RoBERTa Large with a multi-layered perceptron, additionally trained on available target language data as a second few-shot model.
5. XLM-RoBERTa Large with a K-Nearest Neighbour classifier, additionally trained on available target language data as a third few-shot model.

5 Results and Analysis

The final results in the cross-lingual and multi-lingual settings are displayed in Tables 1 and 2 respectively. Each table displays the accuracy on the corresponding test set with the rank achieved by that submission in brackets (out of a total of 87 teams).

In each of the two tables, rows 1, 3 and 2, 4 provide a comparison between the zero-shot and few shot settings. It is interesting to note that few shot learning is *not* effective when using the base version of XLM-RoBERTa. Across both the cross-lingual and multi-lingual settings, the minimal additional data does little to boost performance and in some cases actually reduces performance.

XLM-RoBERTa Large on the other hand, seems to be able to transfer knowledge extracted from training in English to the other languages and *gains the most when those languages are significantly different from English, the language in which the majority of the training data is available in.* The impact of few shot learning is the largest when the difference between the original training data (in this case English) and the target languages is largest. As can be seen from Table 1, the increase on the English-Arabic test set between the zero-shot and few shot settings is *nearly 5 percentage points* despite the few-shot model having been trained on only 8 examples. This same trend can be observed on the English-Chinese dataset albeit to a smaller extent.

The use of a KNN classifier, in place of an MLP, improves performance across the board providing the best results in a lot of the cases and comparable results in the rest. These results seem to validate the results obtained by Snell et al. (2017), who show that the use of a KNN to classify examples related to prototypical examples is an effective method in few-shot learning (Section 2).

6 Conclusions and Future work

This paper described our submission to SemEval 2021 Task2, multi-lingual and cross-lingual word in context disambiguation. Given the nature of the task, wherein we are provided with training data in English and very limited training data in the other target languages, we use zero-shot and few-shot learning.

We hypothesised (Section 1) that two methods will significantly boost performance in the few-shot learning setting: a) The use of "Large" pre-trained models which have been shown to have access to a lot more linguistic information and so generalisability, and b) the use of a KNN classifier instead of a multi-layered perceptron.

Our experiments, described in Section 5, confirm that this is indeed the case. We find that XML-RoBERTa Large is able to significantly increase performance in the few-shot setting, especially when the target languages are dissimilar to English, which is the language the majority of the training data is available in. We additionally find that the use of a KNN classifier boosts performance in the few-shot setting.

We additionally show that when using pre-trained models, tokenisers might split words in ways that are not conducive to the task at hand, especially in languages such as Chinese, where word delimitation is inexact. We handle this limitation by using a comma to delimit words in ways that are specific to the problem, which is both effective and easy to implement.

In future, we intend to explore the use of these methods on other datasets and problems beyond word sense disambiguation.

References

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov. 2015. Siamese neural networks for one-shot image recognition. In *ICML deep learning workshop*, volume 2. Lille.

Sawan Kumar, Sharmistha Jat, Karan Saxena, and Partha Talukdar. 2019. Zero-shot word sense disambiguation using sense definition embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5670–5681, Florence, Italy. Association for Computational Linguistics.

F. Martelli, N. Kalach, G. Tola, and R. Navigli. 2021. SemEval-2021 Task 2: Multilingual and Cross-lingual Word-in-Context Disambiguation (MCL-WiC). In *Proceedings of the 15th Workshop on Semantic Evaluation, 2021.*

T. Mensink, J. Verbeek, F. Perronnin, and G. Csurka. 2013. Distance-based image classification: Generalizing to new classes at near-zero cost. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(11):2624–2637.

George A Miller. 1998. *WordNet: An electronic lexical database*. MIT press.

Maria Pelevina, Nikolay Arefyev, Chris Biemann, and Alexander Panchenko. 2017. Making sense of word embeddings. *CoRR*, abs/1708.03390.

Jake Snell, Kevin Swersky, and Richard S. Zemel. 2017. Prototypical networks for few-shot learning. *CoRR*, abs/1703.05175.

Yaqing Wang, Quanming Yao, James T. Kwok, and Lionel M. Ni. 2020a. Generalizing from a few examples: A survey on few-shot learning. *ACM Comput. Surv.*, 53(3).

Yinglin Wang, Ming Wang, and Hamido Fujita. 2020b. Word sense disambiguation: A comprehensive knowledge exploitation framework. *Knowledge-Based Systems*, 190:105030.

Leiming Yan, Yuhui Zheng, and Jie Cao. 2018. Few-shot learning for short text classification. *Multimedia Tools and Applications*, 77(22):29799–29810.