

# TransWiC at SemEval-2021 Task 2: Transformer-based Multilingual and Cross-lingual Word-in-Context Disambiguation

Hansi Hettiarachchi<sup>♡</sup>, Tharindu Ranasinghe<sup>§</sup>

<sup>♡</sup>School of Computing and Digital Technology, Birmingham City University, UK

<sup>§</sup>Research Group in Computational Linguistics, University of Wolverhampton, UK

`hansi.hettiarachchi@mail.bcu.ac.uk`

`tharindu.ranasinghe@wlv.ac.uk`

## Abstract

Identifying whether a word carries the same meaning or different meaning in two contexts is an important research area in natural language processing which plays a significant role in many applications such as question answering, document summarisation, information retrieval and information extraction. Most of the previous work in this area rely on language-specific resources making it difficult to generalise across languages. Considering this limitation, our approach to SemEval-2021 Task 2 is based only on pretrained transformer models and does not use any language-specific processing and resources. Despite that, our best model achieves 0.90 accuracy for English-English subtask which is very compatible compared to the best result of the subtask; 0.93 accuracy. Our approach also achieves satisfactory results in other monolingual and cross-lingual language pairs as well.

## 1 Introduction

Words' semantics have a dynamic nature which depends on the surrounding context (Pilehvar and Camacho-Collados, 2019). Therefore, the majority of words tends to be polysemous (i.e. have multiple senses). For few examples, words such as "cell", "bank" and "report" can be mentioned. Due to this nature in natural language, it is important to focus on word-in-context sense while extracting the meaning of a word which appeared in a text segment. Also, this is a critical requirement to many applications such as question answering, document summarisation, information retrieval and information extraction.

Word Sense Disambiguation (WSD)-based approaches were widely used by previous research to tackle this problem (Loureiro and Jorge, 2019; Scarlini et al., 2020). WSD associates the word in a text with its correct meaning from a predefined sense inventory (Navigli, 2009). As such

inventories, WordNet (Miller, 1995) and BabelNet (Navigli and Ponzetto, 2012) were commonly used. However, these approaches fail to generalise into different languages as these inventories are often limited to high resource languages. Targeting this gap, SemEval-2021 Task 2: Multilingual and Cross-lingual Word-in-Context Disambiguation is designed to capture the word sense without relying on fixed sense inventories in both monolingual and cross-lingual setting. In summary, this task is designed as a binary classification problem which predicts whether the target word has the same meaning or different meaning in different contexts of the same language (monolingual setting) or different languages (cross-lingual setting).

This paper describes our submission to SemEval-2021 Task 2 (Martelli et al., 2021). Our approach is mainly focused on transformer-based models with different text pair classification architectures. We remodel the default text pair classification architecture and introduce several strategies that outperform the default text pair classification architecture for this task. For effortless generalisation across the languages, we do not use any language-specific processing and resources. In the subtasks where only a few training instances were available, we use few-shot learning and in the subtasks where there were no training instances were available, we use zero-shot learning taking advantage of the cross-lingual nature of the multilingual transformer models.

The remainder of this paper is organised as follows. Section 2 describes the related work done in the field of word-in-context disambiguation. Details of the task data sets are provided in Section 3. Section 4 describes the proposed architecture and Section 5 provides the experimental setup details. Following them, Section 6 demonstrates the obtained results and Section 7 concludes the paper with final remarks and future research directions.

## 2 Related Work

**Unsupervised systems** Majority of the unsupervised WSD systems use external knowledge bases like WordNet (Miller, 1995) and BabelNet (Navigli and Ponzetto, 2012). For each input word, its correct meaning according to the context can be found using graph-based techniques from those external knowledge bases. However, these approaches are only limited to the languages supported by used knowledge bases. More recent works like Hettiarachchi and Ranasinghe (2020a); Ranasinghe et al. (2019a) propose to use stacked word embeddings (Akbik et al., 2018) obtained by general purpose pretrained contextualised word embedding models such as BERT (Devlin et al., 2019) and Flair (Akbik et al., 2019) for unsupervised WSD. Despite their ability to scale over different languages, unsupervised approaches fall behind supervised systems in terms of accuracy.

**Supervised systems** Supervised systems rely on semantically-annotated corpora for training (Raganato et al., 2017; Bevilacqua and Navigli, 2019). Early approaches were based on traditional machine learning algorithms like support vector machines (Iacobacci et al., 2016). With the word embedding-based approaches getting popular in natural language processing tasks, more recent approaches on WSD were based on neural network architectures (Melamud et al., 2016; Raganato et al., 2017). However, they rely on large manually-curated training data to train the machine learning models which in turn hinders the ability of these approaches to scale over unseen words and new languages. More recently, contextual representations of words have been used in WSD where the contextual representations have been employed for the creation of sense embeddings (Peters et al., 2018). However, they also rely on sense-annotated corpora to gather contextual information for each sense, and hence are limited to languages for which gold annotations are available. A very recent approach SensEmBERT (Scarlini et al., 2020) provide WSD by leveraging the mapping between senses and Wikipedia pages, the relations among BabelNet synsets and the expressiveness of contextualised embeddings, getting rid of manual annotations. However, SensEmBERT (Scarlini et al., 2020) only supports five languages making it difficult to use with other languages.

Considering the limitations of the above meth-

ods, in this paper we propose an approach which is based on general purpose transformer models and does not rely on external knowledge bases. Also, our approach shows strong few-shot/zero-shot learning performance removing the hurdle of having manually-curated training data for each language pair.

## 3 Data

The data set used for SemEval-2021 Task 2 is designed targeting a binary classification problem following Pilehvar and Camacho-Collados (2019). To preserve the multilinguality and cross-linguality of the task, five different languages: English, Arabic, French, Russian and Chinese have been considered for data set preparation. In the monolingual setting, per instance, a sentence pair written in the same language is provided with a targeted lemma to predict whether it has the same meaning (True) or different meanings (False) in both sentences. In the cross-lingual setting, each sentence pair is written in two different languages with the same prediction requirement. Few samples from the monolingual and cross-lingual data sets are shown in Table 1.

The monolingual data set covers the language pairs: en-en, ar-ar, fr-fr, ru-ru and zh-zh. For each language, 8-instance trial data sets with labels were provided to give an insight into the task. As training data, 8,000 labelled instances were provided only for the English language and as dev data, 1,000 labelled instances were provided per each language. To use with final evaluation, for each language, 1,000-instance test data sets were provided.

The cross-lingual data set covers the language pairs: en-ar, en-fr, en-ru and en-zh. Similar to the monolingual data set, 8-instance trial data sets with labels were provided for each language pair. However, no training or dev data sets were provided for the cross-lingual setting. To use with the final evaluation, 1,000-instance test data sets were provided per each language pair.

## 4 TransWiC Architecture

The main motivation behind the TransWiC architecture is the success transformer-based architectures had in various natural language processing tasks like offensive language identification (Ranasinghe and Hettiarachchi, 2020; Ranasinghe et al., 2019c; Pitenis et al., 2020), offensive spans identification (Ranasinghe and Zampieri, 2021a; Ranasinghe et al., 2021), language detection (Jauhainen et al.,

	Lang.	Sentence 1	Sentence 2	Label
ML	fr-fr	la <b>souris</b> mange le fromage	le chat court après la <b>souris</b>	T
	en-en	In the private <b>sector</b> , activities are guided by the motive to earn money.	The volume V of the <b>sector</b> is related to the area A of the cap.	F
CL	en-fr	click the right <b>mouse</b> button	le chat court après la <b>souris</b>	F
	en-fr	Any alterations which it is proposed to make as a result of this review are to be <b>reported</b> to the Interdepartmental Committee on Charter Repertory for its approval.	Il a aussi été <b>indiqué</b> que, selon les dossiers médicaux, Justiniano Hurtado Torre était mort de maladie.	T

Table 1: Monolingual (ML) and cross-lingual (CL) sentence pair samples with targeted lemma (highlighted in red colour) and label (T:True, F:False). Lang. column represent the languages which are indicated using ISO 639-1 codes<sup>1</sup>

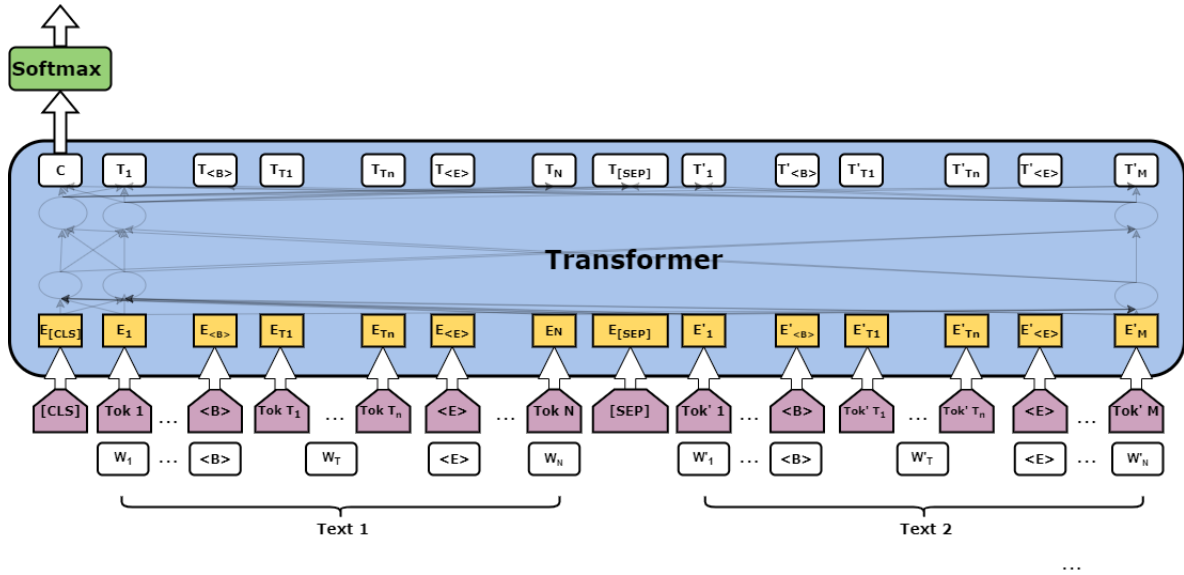


Figure 1: Default sentence pair classification architecture - ([CLS] Strategy).  $W_T$  is the target word.

2021) question answering (Yang et al., 2019) etc. Apart from providing strong results compared to RNN based architectures (Hettiarachchi and Ranasinghe, 2019; Ranasinghe et al., 2019c), transformer models like BERT (Devlin et al., 2019), XLM-R (Conneau et al., 2020) provide pretrained language models that support more than 100 languages. This is a huge benefit when compared to the models like SensEmBERT (Scarlini et al., 2020) which supports only five languages. Furthermore, multilingual and cross-lingual models like multilingual BERT (Devlin et al., 2019) and XLM-R (Conneau et al., 2020) have shown strong transfer learning performance across scarce-resourced languages which can be useful in non-English monolingual subtasks where there are fewer training examples and cross-lingual subtasks where there are no training examples available (Ranasinghe and Zampieri, 2020, 2021b; Ranasinghe et al., 2020a). There-

fore we took the general purpose transformers like BERT (Devlin et al., 2019) and XLM-R (Conneau et al., 2020), reworked their sentence pair classification architecture with so called *strategies* described below to perform well in word-in-context disambiguation task.

**Preprocessing** As a preprocessing step we add two tokens to the transformer model’s vocabulary:  $\langle B \rangle$  and  $\langle E \rangle$ . We place them around the target word in both sentences. For example, the sentence "la souris mange le fromage" with the target word "souris" will be changed to "la  $\langle B \rangle$  souris  $\langle E \rangle$  mange le fromage".

- i [CLS] Strategy - This is the default sentence pair classification architecture with transformers (Devlin et al., 2019) where the two sentences are concatenated with a [SEP] token and passed through a transformer model. Then the

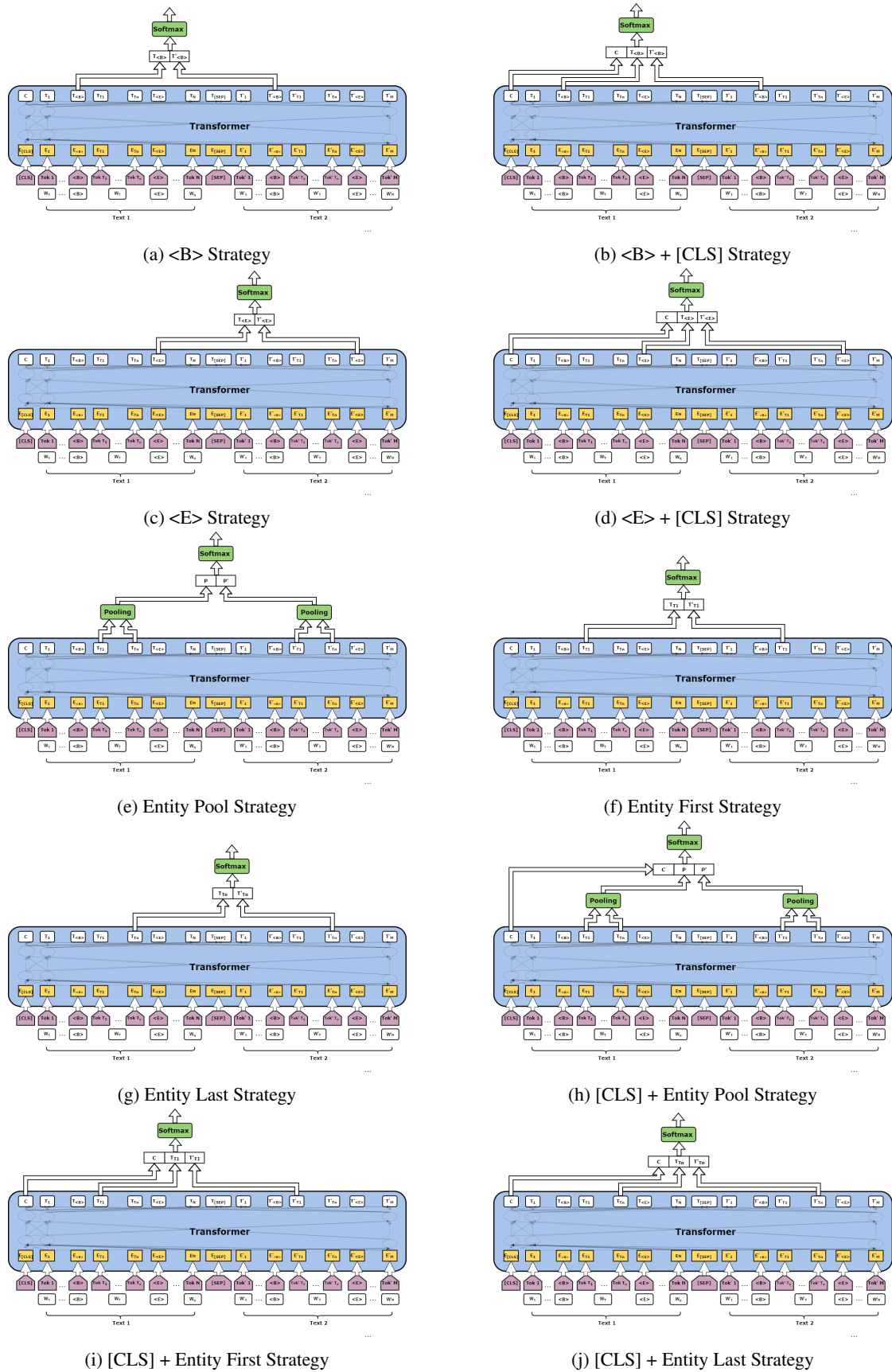


Figure 2: Strategies in the TransWiC Framework.  $W_T$  is the target word.

output of the [CLS] token is fed into a softmax layer to predict the labels (Figure 1).

- ii **<B> Strategy** - We concatenate the output of two <B> tokens of the two sentences and feed it into a softmax layer to predict the labels (Figure 2a).
- iii **<B> + [CLS] Strategy** - We concatenate the output of two <B> tokens of the two sentences with the [CLS] token and feed it into a softmax layer to predict the labels (Figure 2b).
- iv **<E> Strategy** - Output of the two <E> tokens of the two sentences are concatenated and feed into a softmax layer to predict the labels (Figure 2c).
- v **<E> + [CLS] Strategy** - We concatenate the output of two <E> tokens of the two sentences with the [CLS] token and feed it into a softmax layer to predict the labels (Figure 2d).
- vi **Entity Pool Strategy** - To effectively deal with rare words, transformer models use sub-word units or WordPiece tokens as the input to build the models (Devlin et al., 2019). Therefore, there is a possibility that one target word can be separated into several sub-words. In this strategy, we generate separate fixed-length embeddings for each target word by passing its sub-word outputs through a pooling layer. The pooled outputs are concatenated and fed into a softmax layer to predict the labels (Figure 2e).
- vii **Entity First Strategy** - Similar to the previous strategy, instead of using all the sub-words of the target word, we only use the output of the first sub-word in this strategy. We feed the concatenation of these outputs into a softmax layer to predict the labels (Figure 2f).
- viii **Entity Last Strategy** - Similar to the *Entity First Strategy* instead of the first sub-word, we use the last sub-word to represent the target word. We feed their concatenation into a softmax layer to predict the labels (Figure 2g).
- ix **[CLS] + Entity Pool Strategy** - We concatenate the pooled outputs generated by *Entity Pool Strategy* with the [CLS] token and feed it into a softmax layer to predict the labels (Figure 2h).

- x **[CLS] + Entity First Strategy** - Similar to the *[CLS] + Entity Pool Strategy*, instead of the pooled outputs, we concatenate the first sub-word output of the target words with [CLS] token and feed it into a softmax layer to predict the labels (Figure 2i).

- xi **[CLS] + Entity Last Strategy** - In this strategy, we concatenate the last sub-word output of the target words with [CLS] token and feed it into a softmax layer to predict the labels (Figure 2j).

## 5 Experimental Setup

This section describes the training data and hyperparameter configurations used during the experiments.

### 5.1 Training Configurations

**English-English** For the English-English sub-task, we performed training on the English-English training data for each strategy mentioned above. During the training process, the parameters of the transformer model, as well as the parameters of the subsequent layers, were updated. We used the saved model from a particular strategy to get predictions for the English-English test set for that particular strategy.

**Other Monolingual** Since there were less training data available for non-English monolingual datasets, we followed a *few-shot* learning approach mentioned in Ranasinghe et al. (2020c,b). When we are starting the training for non-English monolingual language pairs, rather than training a model from scratch, we initialised the weights saved from the English-English experiment. Then we performed training on the dev data for each language pair separately. Similar to English-English experiments, during the training process, the parameters of the transformer model, as well as the parameters of the subsequent layers, were updated.

**Crosslingual** Since there were no training data available for cross-lingual datasets, we followed a *zero-shot* approach for them. Multilingual and cross-lingual transformer models like multilingual BERT and XLM-R show strong cross-lingual transfer learning performance. They can be trained on one language; typically a resource-rich language and can be used to perform inference on another language. The cross-lingual nature of the transformer models has provided the ability to do this

(Ranasinghe et al., 2020c). Therefore, we used the models trained on the English-English dataset to get predictions for cross-lingual datasets.

## 5.2 Hyperparameter Configurations

We used a Nvidia Tesla K80 GPU to train the models. We divided the input dataset into a training set and a validation set using 0.8:0.2 split. We predominantly fine-tuned the learning rate and the number of epochs of the classification model manually to obtain the best results for the validation set. We obtained  $1e^{-5}$  as the best value for the learning rate and 3 as the best value for the number of epochs. We performed *early stopping* if the validation loss did not improve over 10 evaluation steps. The rest of the hyperparameters which we kept as constants are mentioned in the Appendix. When performing training, we trained five models with different random seeds and considered the majority-class self ensemble mentioned in Hettiarachchi and Ranasinghe (2020b) to get the final predictions.

## 6 Results and Evaluation

Organisers used the accuracy as the evaluation metric as shown in Equation 1 where TP is True Positive, TN is True Negative, FP is False Positive and FN is False Negative.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

Since there were less or no training data available for other monolingual and cross-lingual settings, we trained and evaluated models for each of our strategies using English-English training and dev sets. Then the best models are picked to use with few-shot and zero-shot learning approaches. We report the results obtained by English-English evaluation in Table 2. In the BERT column, we report the results of the bert-large-cased model while in the XLM-R column, we report the results of the xlm-r-large model.

As shown in Table 2, some strategies outperformed the default sentence pair classification architecture. Among all experimented strategies <B> + [CLS] strategy performed best. Usually, multilingual transformer models like XLM-R do not outperform the language-specific transformer models. Surprisingly, in this task XLM-R models outperform bert-large models. We selected three best performing models for the submission; XLM-R

Strategy	BERT	XLM-R
CLS	0.8350	0.7860
<B>	0.8450	0.8750 <sup>‡</sup>
<B> + CLS	0.8590	<b>0.8810<sup>‡</sup></b>
<E>	0.6672	0.5590
<E> + CLS	0.6982	0.5630
Entity Pool	0.8420	0.8521
Entity First	0.8390	0.8462
Entity Last	0.8550	0.8660
CLS + Entity Pool	0.8570	0.8700 <sup>‡</sup>
CLS + Entity First	0.8540	0.8580
CLS + Entity Last	0.8568	0.8610

Table 2: TransWiC accuracy in English-English dev set for each strategy. Best is in Bold. Submitted systems are marked with ‡

<B> + [CLS], XLM-R <B> and XLM-R [CLS] + Entity Pool.

Since multilingual models provided the best results for the English-English dataset, it provided an additional advantage as they can be used directly in other language pairs too as mentioned in Section 5. For other language pairs, we did not perform any evaluation due to the lack of data availability. We trusted the cross-lingual performance of XLM-R and used the best three models of the English-English experiment. For the rest of the monolingual pairs, we used the few-shot learning approach using the given dev sets and for the cross-lingual pairs, we used the zero-shot learning approach mentioned in Section 5.

We report the results we got for the test set in Table 3. According to the results, <B> + [CLS] strategy performs best in all the language pairs except Ar-Ar, where <B> strategy outperforms <B> + [CLS] strategy. When compared to the best models submitted to each language pair, our approach shows very competitive results in the majority of the monolingual language pairs. However, we believe that the cross-lingual performance of our methodology should be improved. Nonetheless, we believe that as a methodology that did not use any language-specific resources and did not see any language-specific data, the results are at a satisfactory level.

## 7 Conclusions

In this paper, we presented our approach for tackling the SemEval-2021 Task 2: Multilingual and Cross-lingual Word-in-Context Disambiguation. We use the pretrained transformer models and re-

Strategy	Monolingual					Crosslingual				
	en-en	ar-ar	fr-fr	ru-ru	zh-zh	en-ar	en-fr	en-ru	en-zh	
I	<B> + [CLS]	<b>0.9040</b>	0.7800	<b>0.7970</b>	<b>0.7610</b>	<b>0.6210</b>	<b>0.6690</b>	<b>0.5860</b>	<b>0.6900</b>	<b>0.7640</b>
	<B>	0.8980	<b>0.7980</b>	0.7760	0.7160	0.6090	0.6260	0.5850	0.6770	0.7280
	[CLS] + Entity Pool	0.8400	0.7621	0.7321	0.6954	0.5880	0.5921	0.5572	0.6561	0.7002
II	Best System	0.9330	0.8480	0.8750	0.8740	0.9100	0.8910	0.8910	0.8940	0.9120

Table 3: Row I shows the accuracy scores for the test set with strategies submitted. Best results for each language pair with our strategies are in bold. Row II shows the accuracy scores for the test set with the best system submitted for each language pair.

model the sentence pair classification architecture for this task with several strategies. Our best strategies outperform the default sentence pair classification setting for English-English. For other monolingual language pairs, we use the few-shot learning approach while for cross-lingual language pairs we use the zero-shot approach. Our results are compatible with the best systems submitted for each language pair and are at a satisfactory level given the fact that we did not use any language-specific processing nor resources.

As future work, we would be looking to improve our results more with new strategies. We would like to experiment with whether adding language-specific processing and resources would improve the results. We are keen to add different neural network architectures like Siamese transformer networks (Reimers and Gurevych, 2019) that perform well in sentence pair classification tasks (Ranasinghe et al., 2019b; Mueller and Thyagarajan, 2016) to the TransWiC framework. Furthermore, we are hoping to work in a multi-task environment and experiment whether transfer learning from a similar task like semantic textual similarity (Cer et al., 2017) would improve the results for this task.

## References

- Alan Akbik, Tanja Bergmann, and Roland Vollgraf. 2019. [Pooled contextualized embeddings for named entity recognition](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 724–728, Minneapolis, Minnesota. Association for Computational Linguistics.
- Alan Akbik, Duncan Blythe, and Roland Vollgraf. 2018. [Contextual string embeddings for sequence labeling](#). In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1638–1649, Santa Fe, New Mexico, USA. Association for Computational Linguistics.
- Michele Bevilacqua and Roberto Navigli. 2019. [Quasi bidirectional encoder representations from transformers for word sense disambiguation](#). In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pages 122–131, Varna, Bulgaria. INCOMA Ltd.
- Daniel Cer, Mona Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. 2017. [SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation](#). In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 1–14, Vancouver, Canada. Association for Computational Linguistics.
- 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.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [BERT: Pre-training of deep bidirectional transformers for language understanding](#). In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.
- Hansi Hettiarachchi and Tharindu Ranasinghe. 2019. [Emoji powered capsule network to detect type and target of offensive posts in social media](#). In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pages 474–480, Varna, Bulgaria. INCOMA Ltd.
- Hansi Hettiarachchi and Tharindu Ranasinghe. 2020a. [BRUMS at SemEval-2020 task 3: Contextualised embeddings for predicting the \(graded\) effect of](#)

- context in word similarity. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 142–149, Barcelona (online). International Committee for Computational Linguistics.
- Hansi Hettiarachchi and Tharindu Ranasinghe. 2020b. **InfoMiner at WNUT-2020 task 2: Transformer-based covid-19 informative tweet extraction**. In *Proceedings of the Sixth Workshop on Noisy User-generated Text (W-NUT 2020)*, pages 359–365, Online. Association for Computational Linguistics.
- Ignacio Iacobacci, Mohammad Taher Pilehvar, and Roberto Navigli. 2016. **Embeddings for word sense disambiguation: An evaluation study**. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 897–907, Berlin, Germany. Association for Computational Linguistics.
- Tommi Jauhiainen, Tharindu Ranasinghe, and Marcos Zampieri. 2021. Comparing approaches to dravidian language identification. In *Proceedings of the 7th Workshop on NLP for Similar Languages, Varieties and Dialects*.
- Daniel Loureiro and Alipio Jorge. 2019. **Liaad at semdeep-5 challenge: Word-in-context (wic)**. In *Proceedings of the 5th Workshop on Semantic Deep Learning (SemDeep-5)*, pages 1–5.
- Federico Martelli, Najla Kalach, Gabriele Tola, and Roberto Navigli. 2021. **SemEval-2021 Task 2: Multilingual and Cross-lingual Word-in-Context Disambiguation (MCL-WiC)**. In *Proceedings of the Fifteenth Workshop on Semantic Evaluation (SemEval-2021)*.
- Oren Melamud, Jacob Goldberger, and Ido Dagan. 2016. **context2vec: Learning generic context embedding with bidirectional LSTM**. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 51–61, Berlin, Germany. Association for Computational Linguistics.
- George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.
- Jonas Mueller and Aditya Thyagarajan. 2016. Siamese recurrent architectures for learning sentence similarity. *AAAI’16*, page 2786–2792. AAAI Press.
- Roberto Navigli. 2009. Word sense disambiguation: A survey. *ACM computing surveys (CSUR)*, 41(2):1–69.
- Roberto Navigli and Simone Paolo Ponzetto. 2012. **Babelnet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network**. *Artificial Intelligence*, 193:217–250.
- Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. **Deep contextualized word representations**. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.
- Mohammad Taher Pilehvar and Jose Camacho-Collados. 2019. **Wic: the word-in-context dataset for evaluating context-sensitive meaning representations**. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1267–1273.
- Zesis Pitenis, Marcos Zampieri, and Tharindu Ranasinghe. 2020. **Offensive language identification in Greek**. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 5113–5119, Marseille, France. European Language Resources Association.
- Alessandro Raganato, Claudio Delli Bovi, and Roberto Navigli. 2017. **Neural sequence learning models for word sense disambiguation**. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1156–1167, Copenhagen, Denmark. Association for Computational Linguistics.
- Tharindu Ranasinghe, Sarthak Gupte, Marcos Zampieri, and Ifeoma Nwogu. 2020a. **Wlv-rit at hasoc-dravidian-codemix-fire2020: Offensive language identification in code-switched youtube comments**. In *In Proceedings of the 12th annual meeting of the Forum for Information Retrieval Evaluation*.
- Tharindu Ranasinghe and Hansi Hettiarachchi. 2020. **BRUMS at SemEval-2020 task 12: Transformer based multilingual offensive language identification in social media**. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1906–1915, Barcelona (online). International Committee for Computational Linguistics.
- Tharindu Ranasinghe, Constantin Orasan, and Ruslan Mitkov. 2019a. **Enhancing unsupervised sentence similarity methods with deep contextualised word representations**. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pages 994–1003, Varna, Bulgaria. INCOMA Ltd.
- Tharindu Ranasinghe, Constantin Orasan, and Ruslan Mitkov. 2019b. **Semantic textual similarity with Siamese neural networks**. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*, pages 1004–1011, Varna, Bulgaria. INCOMA Ltd.
- Tharindu Ranasinghe, Constantin Orasan, and Ruslan Mitkov. 2020b. **TransQuest at WMT2020:**



**Sentence-level direct assessment.** In *Proceedings of the Fifth Conference on Machine Translation*, pages 1049–1055, Online. Association for Computational Linguistics.

Tharindu Ranasinghe, Constantin Orasan, and Ruslan Mitkov. 2020c. **TransQuest: Translation quality estimation with cross-lingual transformers.** In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5070–5081, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Tharindu Ranasinghe, Diptanu Sarkar, Marcos Zampieri, and Alex Ororbia. 2021. **WLV-RIT at SemEval-2021 Task 5: A Neural Transformer Framework for Detecting Toxic Spans.** In *Proceedings of SemEval*.

Tharindu Ranasinghe and Marcos Zampieri. 2020. **Multilingual offensive language identification with cross-lingual embeddings.** In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5838–5844, Online. Association for Computational Linguistics.

Tharindu Ranasinghe and Marcos Zampieri. 2021a. **MUDES: Multilingual Detection of Offensive Spans.** In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*.

Tharindu Ranasinghe and Marcos Zampieri. 2021b. **Multilingual Offensive Language Identification for Low-resource Languages.** *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)*.

Tharindu Ranasinghe, Marcos Zampieri, and Hansi Hettiarachchi. 2019c. **BRUMS at HASOC 2019: Deep learning models for multilingual hate speech and offensive language identification.** In *Proceedings of the 11th annual meeting of the Forum for Information Retrieval Evaluation*.

Nils Reimers and Iryna Gurevych. 2019. **Sentence-BERT: Sentence embeddings using Siamese BERT-networks.** In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Bianca Scarlini, Tommaso Pasini, and Roberto Navigli. 2020. **SensEmBERT: Context-Enhanced Sense Embeddings for Multilingual Word Sense Disambiguation.** *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 8758–8765.

Wei Yang, Yuqing Xie, Aileen Lin, Xingyu Li, Luchen Tan, Kun Xiong, Ming Li, and Jimmy Lin. 2019. **End-to-end open-domain question answering with**

**BERTserini.** In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 72–77, Minneapolis, Minnesota. Association for Computational Linguistics.

## A Appendix

A summary of hyperparameters and their values used to obtain the reported results are mentioned in Table 4. The optimised hyperparameters are marked with ‡ and their optimal values are reported. The rest of the hyperparameter values are kept as constants.

Parameter	Value
learning rate‡	$1e^{-5}$
number of epochs‡	3
adam epsilon	$1e^{-8}$
warmup ration	0.1
warmup steps	0
max grad norm	1.0
max seq. length	120
gradient accumulation steps	1

Table 4: Hyperparameter specifications