The Time-Embedding Travelers at WiC-ITA

Francesco Periti[1], Haim Dubossarsky[2]

[1]University of Milan, Italy
[2]Queen Mary University of London, England

Abstract

The WiC-ITA shared task aims to determine whether a word appearing in two distinct sentences carries the same meaning. The task consists of two subtasks: binary classification (Subtask 1) and ranking (Subtask 2). Each subtask is designed in both a monolingual (Italian) and multilingual (Italian-English) setting. In this report, we present the results of our participation in WiC-ITA. In our experiments, we leverage the condition number of the cosine similarity matrix between XLM-R embeddings and demonstrate competitive performance, ranking among the top positions in both the monolingual and cross-lingual setting. Our results indicate that semantic information is present not only in the last layers but also across the middle layers of XLM-R and throughout the entire architecture. This suggests potential avenues for future research to explore the use of the complete set of embeddings, rather than solely relying on the embeddings extracted from the last layer(s).

Keywords

Word-in-Context, Contextualized Embeddings, Condition Number

1. Introduction

In the last decade, the use of Word Embedding techniques has improved the modeling of lexical semantics. Initially, static embedding models have been employed to encode the dominant semantics of a word into a single vector representation, i.e., word embedding (Mikolov et al., 2013 [1]). However, understanding the meaning of words in their specific contexts is a crucial task for modeling language effectively. This motivated the recent efforts to create contextualized models capable of generating different vector representations according to the context in which the words occur (Devlin et al., 2019 [2]).

Despite the growing popularity of contextualized embeddings in research fields such as Word Sense Disambiguation or Lexical Semantic Shift Detection (Scarlini et al., 2020 [3]; Montanelli and Periti, 2023 [4]), Word-in-Context (WiC) benchmarks that specifically focus on the dynamic of word semantics are relatively recent. The first WiC benchmarks were limited to English (Pilehvar et al., 2019 [5]; Loureiro et al., 2022 [6]). Their success prompted the development of new WiC benchmarks to cover a wider scope of languages (Raganato et al., 2020 [7]; Liu et al., 2021 [8]), test the transfer learning ability in cross-lingual settings (Martelli et al., 2021 [9]), and evaluate graded word similarity in context (Armendariz et al., 2020 [10]).

The WiC-ITA shared task at EVALITA 2023 provides a novel benchmark for evaluating WiC for both a monolingual (L) setting in Italian and a cross-lingual (XL) setting from Italian to English (Cassotti et al., 2023 [11]; Lai et al., 2023 [12]). Inspired by the previous work, WiC-ITA challenges its participants with two sub-tasks:

1. Binary Classification: to establish if a target word w occurring in a pair of sentences $\langle s_1, s_2 \rangle$ has the same meaning or not (Subtask 1);
2. Ranking: to rank the pair of sentences $\langle s_1, s_2 \rangle$ by the degree of similarity of the target word's meaning (Subtask 2).

We participated in both Subtask 1 and 2 as the The Time-Embedding Travelers, alongside three other participants and one baseline system. Each participant was allowed to make three different submissions. In our experiments, we investigated the potential of multilingual pre-trained models in both the L- and XL-WiC setting. Given a pair of sentences $\langle s_1, s_2 \rangle$ and a target word w, our submitted systems compare the word semantics by using the cosine similarity matrix between the XLM-R word embeddings of w extracted from different layers. In particular, we use the condition number of the cosine similarity matrix to assess the degree of semantic similarity between two instances of the word.

In the official ranking, our evaluation phase submission ranked 2nd for the L-Subtask1, 1st for the XL-Subtask1, 2nd for the L-Subtask2, and 1st for the XL-Subtask2. In this paper, we extensively evaluate the effectiveness of our systems on two different multilingual models, namely multilingual BERT (mBERT) and XLM-RoBERTa (XLM-R). Our code is available at https://github.com/FrancescoPeriti/WiC-ITA.

EVALITA 2023: 8th Evaluation Campaign of Natural Language Processing and Speech Tools for Italian, Sep 7 – 8, Parma, IT

francesco.periti@unimi.it (F. Periti); h.dubossarsky@qmul.ac.uk (H. Dubossarsky)
0000-0001-8388-2317 (F. Periti); 0000-0002-2818-6113 (H. Dubossarsky)

2. Background and motivation

BERT is a powerful contextualized model that leverages the Transformer encoder to capture the contextual semantics of words (Devlin et al., 2019 [2]; Vaswani et al., 2017 [13]). Typically, the success of BERT is attributed to its multi-layer (e.g., 12) and multi-head (e.g., 12) self-attention blocks. However, most of the SOTA work only uses the outputs of the final layer(s) (i.e., word embeddings) as input for solving NLP tasks, while ignoring the output of the earlier layers. As a result, the role of different embedding layers for representing the semantics of word occurrences is still unclear. Recently, a limited number of studies have been conducted to explore the nature and characteristics of the BERT embeddings. In particular,Jawahar et al. (2019) [14] indicate that BERT's lower layers capture surface features pertaining to phrase-level information, middle layers capture syntactic features, and higher layers capture semantic features. Devlin et al. (2019) [2] report that combining the last four hidden layers could be beneficial for mainstream tasks such as Named Entitiy Recognition. Ethayarajh (2019) [15] demonstrates that the geometry of the embedding space exhibits anisotropy, meaning that the embeddings of all layers occupy a narrow cone within the vector space. Other work involves probing tasks, as proposed in Hewitt et al. (2019) [16]. These tasks consist of training an auxiliary classifier on top of a model, where the contextualized embeddings serve as features to predict syntactic (e.g., part-of-speech tags) and semantic (e.g., word relations) properties of words. The idea is that if the auxiliary classifier accurately predicts a linguistic property, we can assume that the property is encoded in the tested model. In line with this work, Coenen et al. (2019) [17] investigate the capability of word sense prediction and indicate that earlier-layer embeddings contain significantly more semantic information than conventionally believed.

Thus, our experiments are motivated by the latter finding and inspired by linguistic research that highlights the influential role of morphology and syntax in shaping word meanings (Wysocki and Jenkins, 1987 [18]). In this paper, we challenge the hypothesis that word meanings should be investigated by considering the full output of pre-trained models to encompass not only semantic features of the last layers but also the intricate interplay of semantic, surface, and syntactic features present in the middle and lower layers of the contextualized models.

3. System overview

Our system is a simple threshold-based classifier based on the similarity of two sets of word vectors. In particular, given a pair of sentences $\langle s_1, s_2 \rangle$, and a target word w, we use the output embeddings of a contextualized embedding model to compute a continuous similarity score. This score indicates the extent to which the target w carries the same meaning in the sentences s_1 and s_2.

More precisely, consider a sentence s that contains the word w. Given a contextualized model M, a vector representation of w is extracted from every layer of the model M. This way, the word w in the sentence s is associated with a set of contextualized embeddings denoted by H. It's worth noting that $H \in \mathbb{R}^{n \times d}$, where n is the number of encoders of the model M (e.g., 12) and d is the dimension of the embeddings (e.g., 768). As a result, we denote as H_1 and H_2 the contextualized embeddings of w extracted from the sentences s_1 and s_2, respectively.

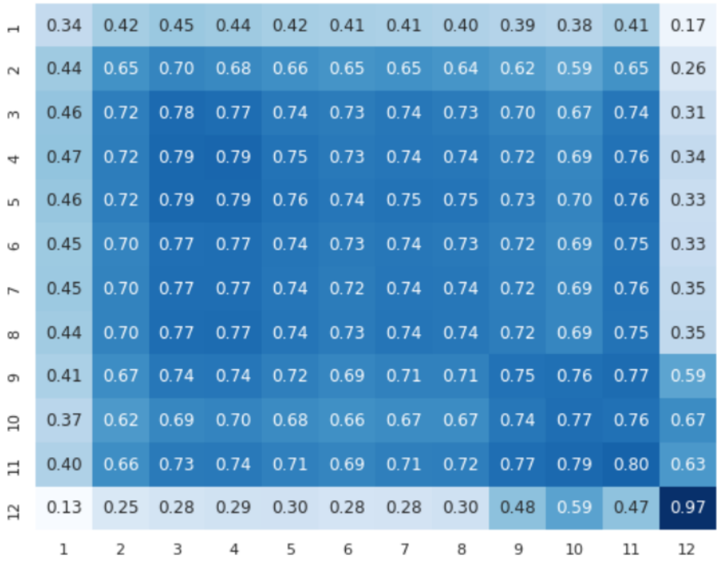
In order to evaluate the similarity of the word w in the contexts s_1 and s_2, we collect the pairwise cosine similarities between H_1 and H_2. We denote as S the similarity matrix between H_1 and H_2 (see Figure 3 as an example [1]) . Our hypothesis is that taking into account information from all layers at once will provide a richer and more comprehensive rapport of the nature of usage similarities of a word between the two sentences. We hypothesize that because many layers are known to capture relevant semantic information, we should consider as many of them as possible together, as they may contain more comprehensive information than a single layer comparison approach.

In order to tap into this pool of similarity scores encoded within S (that contains 144 times more information than a single layer) we use a measure called the *condition number.* The condition number of a matrix, which was already successfully applied in other domains in NLP (Dubossarsky et al., 2020 [19]), provides us with a unified measure that takes into account the many similarities scores between the representations of w of the pair s_1 and s_2 throughout the different layers.

Originally, the condition number of a matrix was used to measure its sensitivity to perturbations, or small changes, in its input. A large condition number indicates that the matrix is ill-conditioned, meaning it is sensitive to small perturbations. On the other hand, a small condition number indicates that the matrix is well-conditioned, meaning that small changes will not affect it much.

In the setting of the WiC task, we interpret the condition number of a similarity matrix as associated with the stability of meaning between the two sentences. Higher similarity scores in S overall indicate two similar word usage and are expected to produce lower (and better) condition number. On the other hand, less similar and more

[1] Corresponding record.
Italian: E siccome mi lascia gps, ecoscandaglio, tutta l'attrezzatura [...] è un *affare*. Fatto. / La rivista nordamericana segnala come presunti sospetti [...] per gli *Affari* Latinoamericani.
English: And since he leaves me gps, depth sounder, all the equipment [...] it's a *bargain*. Done. / The North American magazine lists as alleged suspects [...] for Latin American *affairs*.



	1	2	3	4	5	6	7	8	9	10	11	12
1	0.34	0.42	0.45	0.44	0.42	0.41	0.41	0.40	0.39	0.38	0.41	0.17
2	0.44	0.65	0.70	0.68	0.66	0.65	0.65	0.64	0.62	0.59	0.65	0.26
3	0.46	0.72	0.78	0.77	0.74	0.73	0.74	0.73	0.70	0.67	0.74	0.31
4	0.47	0.72	0.79	0.79	0.75	0.73	0.74	0.74	0.72	0.69	0.76	0.34
5	0.46	0.72	0.79	0.79	0.76	0.74	0.75	0.75	0.73	0.70	0.76	0.33
6	0.45	0.70	0.77	0.77	0.74	0.73	0.74	0.73	0.72	0.69	0.75	0.33
7	0.45	0.70	0.77	0.77	0.74	0.72	0.74	0.74	0.72	0.69	0.76	0.35
8	0.44	0.70	0.77	0.77	0.74	0.73	0.74	0.74	0.72	0.69	0.75	0.35
9	0.41	0.67	0.74	0.74	0.72	0.69	0.71	0.71	0.75	0.76	0.77	0.59
10	0.37	0.62	0.69	0.70	0.68	0.66	0.67	0.67	0.74	0.77	0.76	0.67
11	0.40	0.66	0.73	0.74	0.71	0.69	0.71	0.72	0.77	0.79	0.80	0.63
12	0.13	0.25	0.28	0.29	0.30	0.28	0.28	0.30	0.48	0.59	0.47	0.97

Figure 1: Pairwise similarity matrix (rounded to two decimal places) between the 12 XLM-R embeddings of the target word w in an arbitrary pair $\langle s_1, s_2\rangle$ (the two original sentences appear in footnote 1 above).

varied similarity scores indicate more unrelated usages resulting in a higher (and worse) condition number.

The condition number of a matrix is defined as the multiplication of the matrix's norm by the norm of its reciprocal (i.e., the inverse of the matrix). The norm could be Euclidean norm, Max norm, Frobenius norm, etc. In our experiments, we calculate the condition number (COND) of the similarity matrix S using the Frobenius norm as follows:

$$\text{COND }(S) = \|S\|_F \cdot \|S^{-1}\|_F$$

When we compute the condition number from the similarity matrix S, we assess the degree of semantic similarity sim of a word w in each pair $\langle s_1, s_2\rangle$ as $sim = \text{COND}(S)$ [2].

Furthermore, we also investigate the similarity sim by considering only a subset of S. We test COND$_F$, COND$_M$, and COND$_L$ based on the similarities collected from the first, middle, and last four layers of the model M, respectively.

For the sake of comparison, we set as reference baselines the cosine similarity (CS) of the w embeddings extracted from all the layers of the model M individually, meaning that we compute n different CS scores as

$$\text{CS}_i(H_1[i], H_2[i]) = \frac{H_1[i] \cdot H_2[i]}{\|H_1[i]\|\|H_2[i]\|} \ ,$$

with $i \in 1, ..., n$. Additionally, we compute the cosine similarity CS$_{AVG}$ between the word embeddings obtained by averaging the last four embeddings of H_1 and H_2, respectively.

In line with the WiC-ITA guidelines, we compute the Spearman correlation between the estimated similarity scores and the gold answers. This serves as the evaluation metric for Subtask 2.

In Subtask1, our binary predictions are derived from the similarity scores obtained in Subtask 2. We employ a threshold-based classifier, selecting the threshold value that optimizes the F1 score on the set of sentence pairs used as training set.

4. Experimental setup

In this task, we compared two different contextualized multilingual models, namely mBERT (Devlin et al., 2019 [2]), and XLM-R (Conneau et al., 2020 [20]). We use the Transformers library by HuggingFace to extract contextual word embeddings from mBERT and XLM-R models without performing any fine-tuning stage (Wolf et al., 2020 [21]). We use the base versions, with 12 layers and 768 hidden dimensions: *bert-base-multilingual-cased*, and *xlm-roberta-base*, respectively.

Given a target word w and a pair $\langle s_1, s_2\rangle$. The acquisition of contextual embeddings is done by feeding the models with the sentences s_1 and s_2 individually. For every sentence, we extract the token embedding for the target word w from each layer of the model. Due to the

[2] For ease of interpretation, in our experiment, we utilized the -COND metric. We chose to associate smaller condition numbers with unrelated usages (annotated as 1), while larger numbers with identical usages (annotated as 4).

	Spearman		**Precision**		**Recall**		**F1 score**		**Threshold**
Measures	*Train*	*Test*	*Train*	*Test*	*Train*	*Test*	*Train*	*Test*	*Train*
COND	**0.520**	**0.519**	0.741	0.734	0.742	0.735	0.741	0.734	-6973.703
COND$_M$	0.514	0.513	**0.747**	**0.743**	**0.746**	**0.742**	**0.747**	**0.742**	-1195.522
COND$_L$	0.493	0.493	0.733	0.730	0.736	0.732	0.734	0.730	-972.068
CS$_{10}$	0.408	0.406	0.707	0.701	0.721	0.715	0.708	0.702	0.860
CS$_{11}$	0.404	0.403	0.704	0.698	0.709	0.703	0.706	0.700	0.895
CS$_{AVG}$	0.397	0.395	0.701	0.694	0.714	0.707	0.702	0.694	0.897
CS$_9$	0.395	0.393	0.699	0.693	0.712	0.705	0.701	0.694	0.853
CS$_8$	0.390	0.388	0.694	0.687	0.703	0.695	0.697	0.690	0.839
CS$_7$	0.370	0.367	0.695	0.690	0.698	0.693	0.696	0.691	0.839
CS$_6$	0.369	0.366	0.684	0.676	0.691	0.682	0.686	0.678	0.819
CS$_{12}$	0.342	0.339	0.683	0.679	0.701	0.697	0.686	0.682	0.987
CS$_5$	0.325	0.322	0.666	0.661	0.680	0.675	0.670	0.665	0.816
CS$_4$	0.254	0.252	0.633	0.630	0.657	0.654	0.639	0.636	0.799
COND$_F$	0.226	0.228	0.638	0.634	0.652	0.648	0.643	0.638	-548.832
CS$_3$	0.193	0.191	0.615	0.607	0.652	0.644	0.621	0.614	0.781
CS$_2$	0.156	0.153	0.606	0.597	0.639	0.630	0.613	0.605	0.694
CS$_1$	0.125	0.122	0.605	0.597	0.627	0.618	0.612	0.604	0.503

Table 1
Cross-validation analysis: average scores of WiC-ITA evaluation metrics across 100 different train-test splits. We report in bold the best result for each metric and data set.

byte-pair input encoding scheme employed by BERT-like models, some tokens may not correspond to complete words but rather to word pieces. In such cases, when a word is split into multiple tokens, we build a single word embedding by averaging the embeddings of its constituent word pieces.

Finally, to assess the graded word similarity in the context of a pair of sentences, we calculate similarity scores between the contextualized embeddings of the target word under consideration (See Section 3).

5. Experimental results

In our submissions, we rely on XLM-R as it proved to be more effective than mBERT. To maximize the performance of our system, we leverage the available *train* and *dev* set as a whole. In particular, we randomly generate 100 different train-test splits, with sizes of 2000 and 1305 respectively (equivalent to 60% and 40% of the full dataset). We conduct cross-validation on these 100 splits to validate the use of COND for Subtask2. Additionally, we leverage cross-validation to determine the optimal threshold for Subtask1, meaning that we rely on the average of the 100 best thresholds obtained during cross-validation. The average scores of Spearman correlation, Precision, Recall, and F1 score are presented in Table 1 for each tested measure. For Subtask1 and 2 and for both the L and XL setting, our three submissions correspond to the top three measures based on the F1 score and Spearman correlation, respectively (i.e., COND, COND$_M$, COND$_L$).

For the sake of comparison, Table 2 presents the preliminary performance achieved during the development phase with both XLM-R and mBERT over the Dev and Train sets.

Motivated by the superior results achieved during the development phase, we relied on XLM-R for our final submissions. In particular, we submit the predictions obtained with COND, COND$_M$, COND$_H$. However, in Table 2, it is worth noting that COND also emerged as the leading measure for the mBERT model, proving its consistency. Moreover, we note that for the WiC-ITA task, the embeddings from the last layer of both XLM-R and BERT, as well the embeddings derived by the aggregation of the last four layers, are not as effective as those from other layers. For instance, it is interesting to observe that layer 8 seems to be effective for Subtask1.

In the final evaluation leaderboard for the WiC-ITA task, we ranked 2nd for L-Subtask1, 1st for XL-Subtask1, 2nd for L-Subtask2, and 1st for XL-Subtask2. The leaderboard is reported in Table 3.

Our final results at WiC-ITa demonstrate that COND effectively captures semantic features of word meanings and can be successfully applied to tasks like WiC. Based on our development results, we assert that COND consistently outperforms the CS measure computed over individual contextualized embeddings, for Subtask 1 and 2 in both in L and XL setting. This is particularly interesting considering that CS is commonly utilized in NLP tasks to capture contextual semantics in contextualized embeddings.

Finally, COND$_M$ consistently achieves good results by considering medium layers alone. These results are in line with the findings of Coenen et al. (2019) [17], and suggest that the middle layers of BERT-like models contain valuable information for effectively representing meaning. Therefore, future work should explore the application of COND for WiC and other related NLP tasks such as Lexical Semantic Change Detection (Montanelli

	Development Phase															
	Spearman				**Precision**				**Recall**				**F1 score**			
	XLM-R		mBERT		XLM-R		mBERT		XLM-R		mBERT		XLM-R		mBERT	
Measures	*Dev*	*Train*	*Dev*	*Train*	*Dev*	*Train*	*Dev*	*Train*	*Dev*	*Train*	*Dev*	*Train*	*Dev*	*Train*	*Dev*	*Train*
COND	**0.557**	**0.509**	0.413	**0.406**	0.719	0.748	0.663	0.702	0.698	0.745	0.642	0.702	0.691	0.747	0.630	**0.702**
COND$_M$	0.543	0.501	0.365	0.390	**0.725**	**0.759**	0.635	0.696	**0.710**	**0.752**	0.618	0.702	**0.705**	**0.755**	0.606	0.699
COND$_L$	0.502	0.489	0.275	0.241	0.712	0.744	0.619	0.641	0.688	0.741	0.584	0.663	0.679	0.743	0.551	0.650
CS$_{10}$	0.459	0.394	0.392	0.340	0.660	0.716	0.666	0.690	0.626	0.730	0.624	0.706	0.605	0.721	0.599	0.695
CS$_{11}$	0.439	0.392	0.389	0.331	0.654	0.716	0.666	0.688	0.636	0.716	0.636	0.695	0.625	0.716	0.619	0.691
CS$_{AVG}$	0.447	0.383	0.414	0.360	0.656	0.713	0.674	0.704	0.626	0.726	0.660	0.692	0.607	0.717	0.653	0.697
CS$_9$	0.439	0.382	0.420	0.366	0.666	0.710	**0.682**	0.708	0.636	0.720	0.662	0.699	0.619	0.714	0.652	0.703
CS$_8$	0.435	0.378	**0.429**	0.385	0.667	0.708	0.679	**0.714**	0.640	0.710	**0.664**	**0.704**	0.625	0.709	**0.657**	0.708
CS$_6$	0.425	0.355	0.408	0.337	0.656	0.696	0.679	0.687	0.632	0.697	0.648	0.694	0.617	0.696	0.632	0.691
CS$_7$	0.432	0.354	0.424	0.365	0.686	0.707	0.675	0.702	0.664	0.702	0.658	0.696	0.654	0.704	0.650	0.699
CS$_{12}$	0.387	0.329	0.405	0.353	0.642	0.696	0.670	0.680	0.614	0.710	0.648	0.683	0.594	0.701	0.636	0.682
CS$_5$	0.383	0.312	0.377	0.304	0.643	0.681	0.652	0.674	0.612	0.691	0.622	0.683	0.590	0.685	0.602	0.678
CS$_4$	0.324	0.241	0.329	0.256	0.613	0.648	0.621	0.640	0.578	0.667	0.586	0.661	0.543	0.656	0.553	0.648
COND$_F$	0.180	0.233	0.286	0.245	0.591	0.654	0.617	0.669	0.570	0.661	0.598	0.678	0.544	0.658	0.581	0.673
CS$_3$	0.266	0.182	0.281	0.224	0.602	0.629	0.606	0.631	0.562	0.664	0.578	0.650	0.514	0.640	0.548	0.639
CS$_2$	0.188	0.152	0.212	0.183	0.578	0.618	0.582	0.631	0.552	0.650	0.558	0.653	0.511	0.629	0.523	0.640
CS$_1$	0.138	0.125	0.157	0.166	0.555	0.625	0.591	0.627	0.540	0.636	0.564	0.654	0.506	0.630	0.529	0.637

Table 2
Preliminary performance achieved during the development phase with both XLM-R and mBERT over the Dev and Train sets. We report in bold the best result for each metric, model, and data set.

		Evaluation Phase			
		Subtask1		Subtask2	
Teams	Run	*L-WiC*	*XL-WiC*	*L-WiC*	*XL-WiC*
BERT 4EVER	run1	0.530	0.490	0.340	0.160
BERT 4EVER	run2	0.560	0.520	0.300	0.150
BERT 4EVER	run3	0.560	0.490	-	-
LG	LG	**0.730**	-	0.490	-
The Time-Embedding Travelers	COND$_M$	0.660	0.720	0.520	**0.550**
The Time-Embedding Travelers	COND$_L$	0.620	**0.740**	0.490	0.530
The Time-Embedding Travelers	COND	0.670	0.730	0.550	0.540
extremITA	camoscio lora	0.510	0.540	-	-
extremITA	it5	0.610	0.620	-	-
Baseline	-	0.590	0.560	**0.570**	0.410

Table 3
Final evaluation leaderboard for the WiC-ITA task. The best results are highlighted in bold. Our rankings are as follows: 2nd for L-Subtask1, 1st for XL-Subtask1, 2nd for L-Subtask2, and 1st for XL-Subtask2.

and Periti, 2023 [4]; Tahmasebi et al., 2023 [22]).

Additionally, we observe that the lower layers of XLM-R struggle to distinguish between different meanings when considered individually. However, when incorporated into the standard COND framework, their inclusion leads to improved results compared to COND$_M$. This suggests that our original intuition was correct and that the embedding layers should be considered as a whole, rather than solely focusing on the last layers.

6. Conclusion

Our experiments for the WiC-ITA shared task ranked 2nd for L-Subtask1, 1st for XL-Subtask1, 2nd for L-Subtask2, and 1st for XL-Subtask2. In our submissions, we use the condition number of the cosine similarity matrix between XLM-R embeddings extracted from different layers. Our results support our initial hypothesis that leveraging all the information provided by the pre-trained model can significantly enhance performance on mainstream tasks such as WiC. Specifically, our research suggests that the embedding layers should be considered as a whole, rather than solely focusing on the last layers of the model, which is the conventional practice. This motivates further exploration of alternative measures that can effectively represent word meanings by considering sets of embeddings as input, rather than just individual vectors. Thus, we plan to integrate the condition number into our recently developed approach for Semantic Change Detection [23].

Acknowledgments

This work has in part been funded by the project Towards Computational Lexical Semantic Change Detection supported by the Swedish Research Council (2019–2022; contract 2018-01184), and in part by the research program Change is Key! supported by Riksbankens Jubileumsfond (under reference number M21-0021).

References

[1] T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient Estimation of Word Representations in Vector Space, in: Y. Bengio, Y. LeCun (Eds.), Proc. of ICLR, Scottsdale, Arizona, 2013.

[2] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, in: Proc. of NAACL-HLT, ACL, Minneapolis, Minnesota, 2019, pp. 4171–4186.

[3] B. Scarlini, T. Pasini, R. Navigli, With More Contexts Comes Better Performance: Contextualized Sense Embeddings for All-Round Word Sense Disambiguation, in: Proc. of EMNLP, ACL, Online, 2020, pp. 3528–3539.

[4] S. Montanelli, F. Periti, A Survey on Contextualised Semantic Shift Detection, 2023. arXiv:2304.01666.

[5] M. T. Pilehvar, J. Camacho-Collados, WiC: the Word-in-Context Dataset for Evaluating Context-Sensitive Meaning Representations, in: Proc. of NAACL-HLT, ACL, Minneapolis, Minnesota, 2019, pp. 1267–1273.

[6] D. Loureiro, A. D'Souza, A. N. Muhajab, I. A. White, G. Wong, L. Espinosa-Anke, L. Neves, F. Barbieri, J. Camacho-Collados, TempoWiC: An Evaluation Benchmark for Detecting Meaning Shift in Social Media, in: Proc. of COLING, ICCL, Gyeongju, Republic of Korea, 2022, pp. 3353–3359.

[7] A. Raganato, T. Pasini, J. Camacho-Collados, M. T. Pilehvar, XL-WiC: A Multilingual Benchmark for Evaluating Semantic Contextualization, in: Proc. of EMNLP, ACL, Online, 2020, pp. 7193–7206.

[8] Q. Liu, E. M. Ponti, D. McCarthy, I. Vulić, A. Korhonen, AM2iCo: Evaluating Word Meaning in Context across Low-Resource Languages with Adversarial Examples, in: Proc. of EMNLP, ACL, Punta Cana, Dominican Republic, 2021, pp. 7151–7162.

[9] F. Martelli, N. Kalach, G. Tola, R. Navigli, SemEval-2021 Task 2: Multilingual and Cross-lingual Word-in-Context Disambiguation (MCL-WiC), in: Proc. of SemEval, ACL, Online, 2021, pp. 24–36.

[10] C. S. Armendariz, M. Purver, M. Ulčar, S. Pollak, N. Ljubešić, M. Granroth-Wilding, CoSimLex: A Resource for Evaluating Graded Word Similarity in Context, in: Proc. of LREC, ELRA, Marseille, France, 2020, pp. 5878–5886.

[11] P. Cassotti, L. Siciliani, L. Passaro, M. Gatto, P. Basile, WiC-ITA at EVALITA2023: Overview of the EVALITA2023 Word-in-Context for ITAlian Task, in: Proc. of EVALITA, CEUR.org, Parma, Italy, 2023.

[12] M. Lai, S. Menini, M. Polignano, V. Russo, R. Sprugnoli, G. Venturi, EVALITA 2023: Overview of the 8th Evaluation Campaign of Natural Language Processing and Speech Tools for Italian, in: Proc. of EVALITA, CEUR.org, Parma, Italy, 2023.

[13] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, I. Polosukhin, Attention is All you Need, in: Proc. of NeurIPS, volume 30, CAI, 2017.

[14] G. Jawahar, B. Sagot, D. Seddah, What Does BERT Learn about the Structure of Language?, in: Proc of ACL, ACL, Florence, Italy, 2019, pp. 3651–3657.

[15] K. Ethayarajh, How Contextual are Contextualized Word Representations? Comparing the Geometry of BERT, ELMo, and GPT-2 Embeddings, in: Proc. of EMNLP-IJCNLP, ACL, Hong Kong, China, 2019, pp. 55–65.

[16] J. Hewitt, P. Liang, Designing and Interpreting Probes with Control Tasks, in: Proc. of EMNLP-IJCNLP, ACL, Hong Kong, China, 2019, pp. 2733–2743.

[17] A. Coenen, E. Reif, A. Yuan, B. Kim, A. Pearce, F. Viégas, M. Wattenberg, Visualizing and Measuring the Geometry of BERT, Curran Associates Inc., Red Hook, NY, USA, 2019.

[18] K. Wysocki, J. R. Jenkins, Deriving word meanings through morphological generalization, Reading Research Quarterly (1987) 66–81.

[19] H. Dubossarsky, I. Vulić, R. Reichart, A. Korhonen, The Secret is in the Spectra: Predicting Cross-lingual Task Performance with Spectral Similarity Measures, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2020, pp. 2377–2390.

[20] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, V. Stoyanov, Unsupervised Cross-lingual Representation Learning at Scale, 2020. arXiv:1911.02116.

[21] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. Le Scao, S. Gugger, M. Drame, Q. Lhoest, A. Rush, Transformers: State-of-the-Art Natural Language Processing, in: Proc. of EMNLP, ACL, Online, 2020, pp. 38–45.

[22] N. Tahmasebi, H. Dubossarsky, Computational modeling of semantic change, 2023. arXiv:2304.06337.

[23] F. Periti, A. Ferrara, S. Montanelli, M. Ruskov, What is Done is Done: an Incremental Approach to Semantic Shift Detection, in: Proceedings of the 3rd Workshop on Computational Approaches to Historical Language Change, Association for Computational Linguistics, Dublin, Ireland, 2022, pp. 33–43. URL: https://aclanthology.org/2022.lchange-1.4. doi:10.18653/v1/2022.lchange-1.4.