MULTISEM at SemEval-2020 Task 3: Fine-tuning BERT for Lexical Meaning

Aina Garí Soler
Université Paris-Saclay
CNRS, LIMSI
91400 Orsay, France
aina.gari@limsi.fr

Marianna Apidianaki
Department of Digital Humanities
University of Helsinki
Helsinki, Finland
marianna.apidianaki@helsinki.fi

Abstract

We present the MULTISEM systems submitted to SemEval 2020 Task 3: Graded Word Similarity in Context (GWSC). We experiment with injecting semantic knowledge into pre-trained BERT models through fine-tuning on lexical semantic tasks related to GWSC. We use existing semantically annotated datasets and propose to approximate similarity through automatically generated lexical substitutes in context. We participate in both GWSC subtasks and address two languages, English and Finnish. Our best English models occupy the third and fourth positions in the ranking for the two subtasks. Performance is lower for the Finnish models which are mid-ranked in the respective subtasks, highlighting the important role of data availability for fine-tuning.

1 Introduction

The meaning of words is strongly tied to the context in which they occur: Different contexts might point to different senses or indicate subtler meaning nuances. SemEval 2020 Task 3 "Graded Word Similarity in Context" (GWSC) (Armendariz et al., 2020a) explores the effect of context on meaning, and proposes to predict the similarity of word instances in a continuous, or graded, fashion. GWSC is based on the CoSimLex dataset (Armendariz et al., 2020b) and consists of two subtasks where models have to predict (1) the shift in meaning similarity for a pair of words (w_a, w_b) occurring in different contexts, and (2) the similarity of two word instances in the same context. This is illustrated by sentences c_1 and c_2, two contexts where *dinner* and *breakfast* co-occur.

c_1 (...) After Mickey rings the *dinner* bell, Goofy foolishly leaves the driver's seat for *breakfast*.

c_2 Residence Inns typically feature a complimentary small hot *breakfast* in the morning and a free light *dinner* or snack reception on weekday evenings (...)

A change in meaning similarity occurs between the highlighted words in the two sentences: They are less similar in context c_1 where *dinner* is part of a noun compound (*dinner bell*), than in context c_2 where they describe different kinds of meals. The shift in meaning is reflected in the gold similarity scores assigned to these instance pairs in the GWSC dataset (4.39 vs. 5.35).

We build models for these two subtasks by fine-tuning BERT on existing lexical similarity datasets. Additionally, we propose to approximate similarity of words in context through automatically generated lexical substitutes. We build and evaluate models in two languages, English and Finnish. In Subtask 1, our English and Finnish models ranked third and sixth out of nine participants. In Subtask 2, they are both found at the fourth position among ten participants.[1]

2 Background

Our methodology draws inspiration from recent work on injecting semantic information into pre-trained language models (LMs). This can be done at two stages: during model pre-training or during fine-tuning.

[1] Our code will be made available at https://github.com/ainagari/semeval2020-task3-multisem

Proceedings of the 14th International Workshop on Semantic Evaluation, pages 158–165
Barcelona, Spain (Online), December 12, 2020.

Lauscher et al. (2019) opt for the first, adding an additional lexical task to BERT's two training objectives (language modelling and next sentence prediction) (Devlin et al., 2019). The semantic knowledge used in this additional task comes from pre-defined lexicographic resources (like WordNet (Miller, 1995)), and is shown to be beneficial on almost all tasks in the GLUE benchmark (Wang et al., 2018).

Arase and Tsujii (2019) inject semantic knowledge into BERT by fine-tuning the pre-trained model on paraphrase data. They subsequently fine-tune the model again for the related tasks of paraphrase identification and semantic equivalence assessment, and report results that demonstrate improved performance over a model that has not been exposed to paraphrase data. We follow their approach and fine-tune BERT models for English and Finnish on a set of semantic tasks that are closely related to the GWSC task, since no training data is available for GWSC.

One of our tasks is inspired by the retrofitting approach of Shi et al. (2019). This consists in gathering sentence pairs from the Microsoft Research Paraphrase Corpus (MRPC) (Dolan et al., 2004) that share a word and which are paraphrases of each other (T) or not (F). Shi et al. propose an orthogonal transformation for ELMo (Peters et al., 2018) that is trained to bring representations of word instances closer when they appear in meaning-equivalent contexts. They show that this retrofitting approach improves ELMo's performance in a wide range of semantic tasks at the sentence level (sentiment analysis, inference and sentence relatedness). We follow their data collection method to obtain word instances for fine-tuning BERT. We replace the MRPC with the Opusparcus resource (Creutz, 2018) since it covers two of the languages addressed in GWSC, English and Finnish.

3 System Overview

3.1 Datasets

We fine-tune pre-trained BERT models on semantic tasks that are related to GWSC. We specifically select tasks that address the similarity of word meaning in context, and use the corresponding datasets to make BERT more sensitive to this specific aspect of meaning. Table 1 contains annotated instances from each dataset used in our experiments.

Usim The Usim dataset contains 10 sentences for each of 56 words of different parts of speech, manually annotated with pairwise usage similarity scores (Erk et al., 2009; Erk et al., 2013).[2] As in GWSC, similarity scores are graded and range from 1 (completely different) to 5 (same meaning). The Usim sentences come from the SemEval 2007 Lexical Substitution task dataset (McCarthy and Navigli, 2007).[3] To binarize the usage similarity scores and use them for fine-tuning, we consider sentence pairs annotated with low similarity scores (score < 2) as instances denoting a different meaning (F), and highly similar sentence pairs (score > 4) as instances of the same sense (T). In total, we use 1,399 Usim sentence pairs for fine-tuning.

Concepts in Context (CoInCo) The CoInCo corpus (Kremer et al., 2014) contains manually selected substitutes for all content words in a sentence. Substitute overlap between different word instances reflects their semantic similarity: instance pairs with similar meaning share a higher number of substitutes. We binarize the data as in Garí Soler et al. (2019) by assigning instance pairs to a class describing the same (T) or different (F) meaning depending on their shared substitutes. The data sample used by Garí Soler et al. contains instances with at least four substitutes: T pairs involve instances that have at least 75% of substitutes in common, and F pairs correspond to instances with no substitute overlap. We gather additional data from CoInCo by relaxing the class inclusion constraints. Specifically, we retain all instances regardless of the number of available substitutes. We consider as T examples instance pairs that have at least 50% of substitutes in common, and as F examples pairs that share at most one substitute.

We retain up to 500 instance pairs per CoInCo lemma, when available. We balance the two classes (T and F) and merge the obtained instances with Garí Soler et al. (2019)'s dataset (5,023 pairs) removing the duplicates. In total, we have 22,226 CoInCo instance pairs for fine-tuning. We use these instances in combination with the Usim data.

[2] http://www.dianamccarthy.co.uk/downloads/WordMeaningAnno2012/

[3] http://www.dianamccarthy.co.uk/task10index.html

Class	Sentence 1	Sentence 2
Usim		
T (4.3)	We recommend that you **check** with us beforehand.	I have **checked** multiple times with my order and that is not the case.
F (1.3)	The romance is uninspiring... and **dry**.	If the mixture is too **dry**, add some water; if it is too soft, add some flour.
CoInCo		
T	A mission to end a **war** {*fight, battle, conflict, combat, struggle, crusade*}	He knew the **war** would soon be over and he would be heading home. {*fight, battle, conflict, combat, bloodshed, fighting, hostility*}
F	You're all **right**? {*ok, okay, well, safe, good*}	He's sitting **right** there at the bar! {*over, straight, exactly, direcly, just, precisely, currently*}
WiC		
T	Laws limit the **sale** of handguns .	They tried to boost **sales**.
F	She didn't want to **answer**.	This may **answer** her needs.
ukWaC-subs		
a (T)	For neuroscientists, the message was **clear**.	For neuroscientists, the message was **unambiguous**.
b (F)	Need a **present** for someone with a unique name?	Need a **moment** for someone with a unique name?
c (F')	Overdue tasks display on the due **date**.	Overdue tasks display on the due **heritage**.
Opusparcus		
T	I **love** you so much	I **love** you to the moon and back.
F	yes, Mary, I would **love** to dance.	Why do I **love** him?

Table 1: Examples of instance pairs from each dataset used for fine-tuning.

Word-in-Context (WiC) The WiC dataset contains pairs of word instances in context with the same or a different meaning (Pilehvar and Camacho-Collados, 2019). Sentences come from WordNet (Fellbaum, 1998), VerbNet (Schuler, 2006) and Wiktionary examples, and were automatically annotated based on information provided in these resources. We use the training set (5,428 sentence pairs) with its labels (T or F) as data for fine-tuning.

ukWaC-subs The GWSC task addresses pairs of different words that can have similar meanings in some contexts and not in others (e.g., *room* and *cell*). Given that no training data is available, we automatically create one more dataset for fine-tuning called ukWaC-subs, which approximates this task.

ukWaC-subs contains pairs of sentences (p_1, p_2) that differ in one word only. We create data by substituting a word w in p_1 by either (a) a correct substitute; (b) a word that is a good synonym of w and could have been a correct substitute in another context but not in this one; or (c) a random word of the same part of speech as w. This is illustrated by the three ukWaC-subs sentences in Table 1. With (a), we expect BERT to learn that *clear* is being used in its *unambiguous* sense in this context. In (b), we tell BERT that despite the (out-of-context) similarity between *present* and *moment*, the latter is not adequate in this context. With (c), we want BERT to learn to distinguish *date* from a completely unrelated word (*heritage*). We use this data for a 3-way classification task.

We create this dataset by gathering sentences from the ukWaC corpus (Baroni et al., 2009) and automatically annotating them with lexical substitutes. We identify the content words in a sentence and use as their candidate substitutes their paraphrases in the Paraphrase Database (PPDB) lexical XXL package (Ganitkevitch et al., 2013; Pavlick et al., 2015).[4] The PPDB resource was automatically constructed by a bilingual pivoting method. Every paraphrase pair has a PPDB 2.0 score indicating its quality. We only consider as candidates for substitution pairs with a score above 2. We then use the context2vec lexical substitution model (Melamud et al., 2016) to rank the candidates according to how

[4] http://paraphrase.org/

well they fit in a context. context2vec is a biLSTM model that jointly learns static representations of words and dynamic context representations. We rank candidate substitutes using the following formula:

$$c2v\_score = \frac{cos(s,t)+1}{2} \times \frac{cos(s,C)+1}{2} \quad (1)$$

where s is the static representation of the candidate substitute, C is the context embedding of the sentence and t is the static embedding of a word instance i we want to replace. Using this formula, we obtain an ordered ranking R of substitutes for an instance i in context C. The highest-ranked substitute is viewed as correct and serves to create instances of type (a). A random word of the same part of speech found in the corpus makes an instance of class (c). To obtain instances of class (b) we could in principle take the last substitute in the ranking. However, due to the noise that exists in PPDB, these often are not correct paraphrases of the target word. We therefore apply a filtering strategy proposed by Garí Soler et al. (2019) which checks whether substitutes in adjacent positions (s_i, s_{i+1}) in the ranking R form a paraphrase pair in PPDB. If this is not the case for a specific pair, we stop checking and retain s_{i+1} as a substitute that represents a different meaning of the target word.

Once the substitutes have been collected, 40% of the instances are assigned to class (a), 30% to class (b) and 30% to (c). One sentence may contain more than one training instance if a substitute ranking is available for different words in it. A training instance is created by replacing the word with the substitute required by the class it has been assigned to. We create 100,000 instances that we use to fine-tune BERT.

OpusParcus Shi et al. (2019) show that retrofitting ELMo with paraphrases improves its performance on lexical semantic tasks. We follow a similar approach and use paraphrases to fine-tune BERT before applying it to GWSC. We use paraphrases from the Open Subtitles Paraphrase Corpus (Opusparcus) (Creutz, 2018). We use this corpus instead of the Microsoft Research Paraphrase Corpus (Dolan et al., 2004) used by Shi et al. (2019), because it contains paraphrase pairs for six European languages, including English and Finnish which are addressed in GWSC.

Paraphrase pairs in Opusparcus were extracted from movies and TV shows subtitles, and are ranked by quality. We use paraphrases from the Opusparcus training set with a quality score higher than 15,[5] and create our own training instances following the procedure of Shi et al. (2019). Every pair of paraphrases that share a content word constitutes a positive example (T). For every T, we create a negative example (F) by selecting a pair of sentences from the resource that share the same word but are not paraphrases of each other. To avoid creating examples for target words that are highly frequent and have fuzzy semantics, we omit instances of the 200 most frequent words in the Google Books NGram corpus (Michel et al., 2011) (e.g., *make*, *get*, *good*). In total, we use 100,000 sentence pairs for fine-tuning the English model and 60,520 for Finnish.

3.2 Models

We use these five datasets to fine-tune pre-trained BERT models for English and Finnish. All tasks require comparing the meaning of word instances in two different sentences. We form an input sequence (sentence pair) for BERT by joining the two sentences together with the separator token (`[SEP]`) in between. Since the task is at the word level, we do not build our classifier on top of the `[CLS]` token which is an aggregation of the whole input sequence. Instead, our classifier receives as input the BERT representations of the target word instances at the last layer. BERT uses wordpiece tokenization (Wu et al., 2016) which means that a target word may be split into several tokens. For words that have been split, we average the representations of each wordpiece. We use two kinds of heads for fine-tuning.

- **Classification head**: The representations of the two target tokens are concatenated and fed to a linear classifier which outputs probabilities for each class. We use a cross entropy loss for training. We call this head CLASSIF.
- **Cosine Distance head**: We apply the Cosine Embedding Loss (PyTorch (Paszke et al., 2019)) to the representations of the two target tokens at the last layer. This loss increases the cosine distance of

[5] Scores range from ∼77 (best quality) to ∼2 (worst quality).

two tokens if they do not have the same meaning, and decreases it otherwise. We refer to this head as COSDIST.

Note that the ukWaC-subs dataset is compatible with the CLASSIF head only because it has three classes. To predict the similarity of two target tokens in the GWSC task, we extract their representations from the different layers of a fine-tuned model. We use cosine similarity ($cossim$) as our similarity metric. In Subtask 2, which consists in predicting the similarity scores for a pair of words (w_a, w_b) in the same context, we simply calculate the cosine similarity of their representations in a specific layer. In Subtask 1, we need to predict a change in similarity between two words w_a and w_b in two different contexts (c_1, c_2). We estimate the change in similarity (ΔSim) with a simple subtraction:

$$\Delta Sim = cossim(w_{a_{c2}}, w_{b_{c2}}) - cossim(w_{a_{c1}}, w_{b_{c1}}) \qquad (2)$$

where $w_{a_{c2}}$ is the representation of word w_a in context c_2.

3.3 Experimental Setup

We participate in Subtasks 1 and 2 for English and Finnish.[6] For English, we fine-tune the `bert-base-uncased` model. For Finnish, we use the uncased Finnish model (`finnish`) (Virtanen et al., 2019)[7] and the uncased Multilingual BERT-base model (`multilingual`).[8] For faster fine-tuning, we set the maximum length to 128 wordpiece and omit examples where a target word occurs after this position.

As a development set for English, we use the officially released GWSC trial data (10 sentence pairs) and an earlier release of trial data (8 sentence pairs), both distinct from the test set. We use these data to select the best models and hyperparameters for our official submissions to GWSC. The English test set consists of 340 context pairs for Subtask 1 and 680 unique contexts for Subtask 2. We fine-tune `bert-base-uncased` separately on each of our English datasets experimenting with the two classification heads {CLASSIF, COSDIST} and with different learning rates {5e-5, 1e-6, 1e-7} for up to 15 epochs. These hyperparameters, along with the layer the word representations are extracted from, are set on the GWSC trial data. Our submitted models were fine-tuned on WiC, Opusparcus and CoInCo-Usim with a learning rate of 5e-5 and 0.1 dropout for 4, 3 and 2 epochs, respectively. The ukWaC-subs model was fine-tuned for 11 epochs with a learning rate of 1e-6 and 0.2 dropout. Dropout was determined based on results on 2,000 held-out ukWaC-subs instances. Since no trial dataset was released for Finnish, we fixed the hyperparameters for our models to those that worked best for the English Opusparcus data. Our submitted predictions are from the higher layers of the models fine-tuned with the CLASSIF head. The test set for Finnish consists of 24 context pairs for Subtask 1 (48 unique contexts for Subtask 2).[9]

The metrics used to evaluate model predictions are the uncentered Pearson correlation (ρ) in Subtask 1, and the harmonic mean of Pearson and Spearman correlations ($\bar{\rho}$) in Subtask 2.

4 Results

Results for the two English and Finnish subtasks are presented in Table 2. We report results of the two best systems submitted to each subtask (marked with †) along with results calculated during the post-evaluation phase for comparison. These include baseline predictions made by BERT models without fine-tuning.

Although the two subtasks are highly related, different models perform best in each one. For English, the best result in Subtask 1 (among our official submissions) is obtained by the model fine-tuned on WiC data with the COSDIST head ($\rho = 0.760$) which occupies the third position in the final ranking. It is closely followed by the model fine-tuned on paraphrase data with the CLASSIF head. The best performing model in Subtask 2 is the one fine-tuned on the ukWaC-subs data ($\bar{\rho} = 0.718$) which ranked fourth. The second best model uses the COSDIST head and is trained on the CoInCo and Usim data together. All

[6] We did not address Croatian and Slovenian due to the lack of datasets that could be used for fine-tuning.

[7] `https://github.com/TurkuNLP/FinBERT`

[8] `https://github.com/google-research/bert/blob/master/multilingual.md`

[9] We use the HuggingFace `transformers` library (Wolf et al., 2019) to implement our experiments.

Model	Subtask 1	Subtask 2
English		
WiC COSDIST	† 0.760$_{11}$	0.689$_{11}$
ukWaC-subs	0.751$_{10}$	† **0.718**$_{10}$
Opusparcus CLASSIF	† 0.751$_{11}$	0.669$_{6}$
CoInCo + Usim COSDIST	**0.765**$_{10}$	† 0.686$_{6}$
`bert-base-uncased`	0.715$_{11}$	0.661$_{11}$
Finnish		
`multilingual` Opusparcus CLASSIF	† 0.593$_{9}$	† 0.192$_{11}$
`multilingual` Opusparcus CLASSIF	**0.718**$_{6}$	0.286$_{5}$
`finnish` Opusparcus CLASSIF	† 0.500$_{12}$	† 0.491$_{9}$
`finnish` Opusparcus CLASSIF	0.550$_{1}$	0.568$_{3}$
`multilingual`	0.677$_{11}$	0.388$_{9}$
`finnish`	0.577$_{12}$	**0.671**$_{12}$

Table 2: Results of our English and Finnish models in GWSC Subtasks 1 and 2. The models are compared to three BERT-based baselines without fine-tuning. The evaluation metric in Subtask 1 is Pearson's correlation coefficient. In Subtask 2, it is the harmonic mean of Pearson and Spearman's correlation coefficients. Our official submissions to the GWSC task for each language are marked with †. Subscripts indicate the BERT model layer used.

English models outperform the BERT-based baseline without fine-tuning ($\rho = 0.715$ and $\bar{\rho} = 0.661$). This demonstrates the higher quality of lexical semantic knowledge in our fine-tuned models.

Best results for the Finnish Subtasks 1 and 2 are also produced by different models. The `multilingual` model performs better on Subtask 1 and the `finnish` model on Subtask 2. We observe that the `multilingual` model tends to assign very high similarities to all word instance pairs, which explains its low performance in Subtask 2. At the same time, however, it does well on Subtask 1 because it captures the magnitude of the difference in similarity between two pairs. Given that no trial data (development set) are available for Finnish and that the maximum number of submissions was nine, we could only try at most five layers per model at submission time. The models were ranked sixth and fourth in Subtasks 1 and 2.

During the post-evaluation phase, we had the possibility to test all layers of the models. The sixth layer of the `multilingual` model fine-tuned on Finnish Opusparcus data outperforms the `multilingual` baseline on Subtask 1 ($\rho = 0.718$ vs $\rho = 0.677$), but the other fine-tuned models do not improve over their respective baselines. Surprisingly, the `finnish` baseline model in Subtask 2 ($\bar{\rho} = 0.671$) outperforms the top-ranked model for Finnish among all teams that participated in the task ($\bar{\rho} = 0.645$).

5 Conclusion

We have participated in the SemEval task Graded Word Similarity in Context for English and Finnish, with models integrating different notions of word similarity. We have specifically investigated the effect of fine-tuning pre-trained BERT models on existing datasets that address word meaning similarity in context. Furthermore, we have proposed a new fine-tuning task where in-context lexical similarity is approximated through automatic substitute annotations.

Our English models are ranked at the third and fourth position in the two GWSC subtasks, and outperform a BERT-based baseline without fine-tuning. This demonstrates the benefit of fine-tuning BERT on a task that is closely related to the end task, even when the data used for fine-tuning are automatically obtained. Due to the scarcity of resources for Finnish, we could only fine-tune models with paraphrases. The Finnish models are mid-ranked among all participating systems.

Acknowledgements

This work has been supported by the French National Research Agency under project ANR-16-CE33-0013. The work is also part of the FoTran project, funded by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement № 771113).

References

Yuki Arase and Jun'ichi Tsujii. 2019. Transfer fine-tuning: A BERT case study. 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 5393–5404, Hong Kong, China, November. Association for Computational Linguistics.

Carlos S. Armendariz, Matthew Purver, Senja Pollak, Nikola Ljubešić, Matej Ulčar, Marko Robnik-Šikonja, Ivan Vulić, and Mohammad Taher Pilehvar. 2020a. SemEval-2020 Task 3: Graded Word Similarity in Context (GWSC). In *Proceedings of the 14th International Workshop on Semantic Evaluation*.

Carlos S. Armendariz, Matthew Purver, Matej Ulčar, Senja Pollak, Nikola Ljubešić, and Mark Granroth-Wilding. 2020b. CoSimLex: A Resource for Evaluating Graded Word Similarity in Context. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 5878–5886, Marseille, France, May. European Language Resources Association.

Marco Baroni, Silvia Bernardini, Adriano Ferraresi, and Eros Zanchetta. 2009. The WaCky Wide Web: A Collection of Very Large Linguistically Processed Web-Crawled Corpora. *Journal of Language Resources and Evaluation*, 43(3):209–226.

Mathias Creutz. 2018. Open Subtitles Paraphrase Corpus for Six Languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, May. European Language Resources Association (ELRA).

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, June. Association for Computational Linguistics.

Bill Dolan, Chris Quirk, and Chris Brockett. 2004. Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources. In *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*, pages 350–356, Geneva, Switzerland, August. COLING.

Katrin Erk, Diana McCarthy, and Nicholas Gaylord. 2009. Investigations on Word Senses and Word Usages. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 10–18, Suntec, Singapore, August. Association for Computational Linguistics.

Katrin Erk, Diana McCarthy, and Nicholas Gaylord. 2013. Measuring Word Meaning in Context. *Computational Linguistics*, 39(3):511–554.

Christiane Fellbaum, editor. 1998. *WordNet: An Electronic Lexical Database*. Language, Speech, and Communication. MIT Press, Cambridge, MA.

Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. 2013. PPDB: The Paraphrase Database. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 758–764, Atlanta, Georgia. Association for Computational Linguistics.

Aina Garí Soler, Marianna Apidianaki, and Alexandre Allauzen. 2019. Word usage similarity estimation with sentence representations and automatic substitutes. In *Proceedings of the Eighth Joint Conference on Lexical and Computational Semantics (*SEM 2019)*, pages 9–21, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Gerhard Kremer, Katrin Erk, Sebastian Padó, and Stefan Thater. 2014. What Substitutes Tell Us - Analysis of an "All-Words" Lexical Substitution Corpus. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 540–549, Gothenburg, Sweden, April. Association for Computational Linguistics.

Anne Lauscher, Ivan Vulić, Edoardo Maria Ponti, Anna Korhonen, and Goran Glavaš. 2019. Informing unsupervised pretraining with external linguistic knowledge. *arXiv preprint arXiv:1909.02339*.

Diana McCarthy and Roberto Navigli. 2007. SemEval-2007 Task 10: English Lexical Substitution Task. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 48–53, Prague, Czech Republic. Association for Computational Linguistics.

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.

Jean-Baptiste Michel, Yuan Kui Shen, Aviva Presser Aiden, Adrian Veres, Matthew K Gray, Joseph P Pickett, Dale Hoiberg, Dan Clancy, Peter Norvig, Jon Orwant, et al. 2011. Quantitative analysis of culture using millions of digitized books. *Science*, 331(6014):176–182.

George A. Miller. 1995. WordNet: A Lexical Database for English. *Commun. ACM*, 38(11):39–41.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

Ellie Pavlick, Pushpendre Rastogi, Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. 2015. PPDB 2.0: Better paraphrase ranking, fine-grained entailment relations, word embeddings, and style classification . In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 425–430, Beijing, China. Association for Computational Linguistics.

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 José Camacho-Collados. 2019. WiC: 10, 000 Example Pairs for Evaluating Context-Sensitive Representations. *Accepted at the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Karin Kipper Schuler. 2006. *VerbNet: A Broad-Coverage, Comprehensive Verb Lexicon*. Ph.D. thesis, University of Pennsylvania.

Weijia Shi, Muhao Chen, Pei Zhou, and Kai-Wei Chang. 2019. Retrofitting contextualized word embeddings with paraphrases. 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 1198–1203, Hong Kong, China, November. Association for Computational Linguistics.

Antti Virtanen, Jenna Kanerva, Rami Ilo, Jouni Luoma, Juhani Luotolahti, Tapio Salakoski, Filip Ginter, and Sampo Pyysalo. 2019. Multilingual is not enough: BERT for Finnish. *arXiv preprint arXiv:1912.07076*.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium, November. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R'emi Louf, Morgan Funtowicz, and Jamie Brew. 2019. HuggingFace's Transformers: State-of-the-art Natural Language Processing. *arXiv preprint arXiv:1910.03771*.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Łukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2016. Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation. *arXiv preprint arXiv:1609.08144*.