

WiC: the Word-in-Context Dataset for Evaluating Context-Sensitive Meaning Representations

Mohammad Taher Pilehvar^{1,2} and Jose Camacho-Collados³

¹DTAL, University of Cambridge, UK

²Tehran Institute for Advanced Studies (TeIAS), Tehran, Iran

³School of Computer Science and Informatics, Cardiff University, UK
mp792@cam.ac.uk, camachocolladosj@cardiff.ac.uk

Abstract

By design, word embeddings are unable to model the dynamic nature of words' semantics, i.e., the property of words to correspond to potentially different meanings. To address this limitation, dozens of specialized meaning representation techniques such as sense or contextualized embeddings have been proposed. However, despite the popularity of research on this topic, very few evaluation benchmarks exist that specifically focus on the dynamic semantics of words. In this paper we show that existing models have surpassed the performance ceiling of the standard evaluation dataset for the purpose, i.e., Stanford Contextual Word Similarity, and highlight its shortcomings. To address the lack of a suitable benchmark, we put forward a large-scale Word in Context dataset, called WiC, based on annotations curated by experts, for generic evaluation of context-sensitive representations. WiC is released in <https://pilehvar.github.io/wic/>.

1 Introduction

One of the main limitations of mainstream word embeddings lies in their static nature, i.e., a word is associated with the same embedding, independently from the context in which it appears. Therefore, these embeddings are unable to reflect the dynamic nature of ambiguous words¹, in that they can correspond to different (potentially unrelated) meanings depending on their usage in context (Camacho-Collados and Pilehvar, 2018). To get around this limitation dozens of proposals have been put forward, mainly in two categories: multi-prototype embeddings (Reisinger and Mooney, 2010; Neelakantan et al., 2014; Pelevina et al., 2016), which usually leverage context clustering in order to learn distinct representations for individual meanings of words, and contextualized

¹Ambiguous words are important as they constitute the most frequent words in a natural language (Zipf, 1949).

word embeddings (Melamud et al., 2016; Peters et al., 2018), which instead compute a single dynamic embedding for a given word which can adapt itself to arbitrary contexts for the word.

Despite the popularity of research on these specialised embeddings, very few benchmarks exist for their evaluation. Most works in this domain either perform evaluations on word similarity datasets (in which words are presented in isolation; hence, they are not suitable for verifying the dynamic nature of word semantics) or carry out impact analysis in downstream NLP applications (usually, by taking word embeddings as baseline). Despite providing a suitable means of verifying the effectiveness of the embeddings, the downstream evaluation cannot replace generic evaluations as it is difficult to isolate the impact of embeddings from many other factors involved, including the algorithmic configuration and parameter setting of the system. To our knowledge, the Stanford Contextual Word Similarity (SCWS) dataset (Huang et al., 2012) is the only existing benchmark that specifically focuses on the dynamic nature of word semantics.² In Section 4 we will explain the limitations of this dataset for the evaluation of recent work in the literature.

In this paper we propose WiC, a novel dataset that provides a high-quality benchmark for the evaluation of context-sensitive word embeddings. WiC provides multiple interesting characteristics: (1) it is suitable for evaluating a wide range of techniques, including contextualized word and sense representation and word sense disambiguation; (2) it is framed as a binary classification dataset, in which, unlike SCWS, identical words are paired with each other (in different con-

²With a similar goal in mind but focused on hypernymy, Vyas and Carpuat (2017) developed a benchmark to assess the capability of automatic systems to detect hypernymy relations in context.

| | |
|----------|--|
| F | There’s a lot of trash on the <i>bed</i> of the river — I keep a glass of water next to my <i>bed</i> when I sleep |
| F | <i>Justify</i> the margins — The end <i>justifies</i> the means |
| T | <i>Air</i> pollution — Open a window and let in some <i>air</i> |
| T | The expanded <i>window</i> will give us time to catch the thieves — You have a two-hour <i>window</i> of clear weather to finish working on the lawn |

Table 1: Sample positive (T) and negative (F) pairs from the WiC dataset (target word in *italics*).

texts); hence, a context-insensitive word embedding model would perform similarly to a random baseline; and (3) it is constructed using high quality annotations curated by experts.

2 WiC: the Word-in-Context dataset

We frame the task as binary classification. Each *instance* in WiC has a target word w , either a verb or a noun, for which two contexts, c_1 and c_2 , are provided. Each of these contexts triggers a specific meaning of w . The task is to identify if the occurrences of w in c_1 and c_2 correspond to the same meaning or not. Table 1 lists some examples from the dataset. In what follows in this section, we describe the construction procedure of the dataset.

2.1 Construction

Contextual sentences in WiC were extracted from example usages provided for words in three lexical resources: (1) WordNet (Fellbaum, 1998), the standard English lexicographic resource; (2) VerbNet (Kipper-Schuler, 2005), the largest domain-independent verb-based resource; and (3) Wiktionary³, a large collaborative-constructed online dictionary. We used WordNet as our core resource, exploiting BabelNet’s mappings (Navigli and Ponzetto, 2012) as a bridge between Wiktionary and VerbNet to WordNet. Lexicographer examples constitute a reliable base for the construction of the dataset, as they are curated in a way to be clearly distinguishable across different senses of a word.

2.1.1 Compilation

As explained above, the dataset is composed of instances, each of which contain a target word and two examples containing the target word. An instance can be either positive or negative, depending on whether the corresponding c_1 and c_2 are listed for the same sense of w in the target resource. In order to compile the dataset, we first

³<https://www.wiktionary.org/>

obtained all the possible positive and negative instances from all resources, with the only condition of the surface word form occurring in both c_1 and c_2 .⁴ The total number of initial examples extracted from all resources at this stage were 23,949, 10,564 and 636 for WordNet, Wiktionary and VerbNet, respectively. We first compiled the test and development sets with two constraints: (1) not having more than three instances for the same target word, and (2) not having repeated contextual sentences across instances. These constraints were enforced to have a diverse and balanced set which covers as many unique words as possible. With all these constraints in mind, we set apart 1,600 and 800 instances for the test and development sets, respectively. We ensured that all the splits were balanced for their positive and negative examples. The remaining instances whose examples did not overlap with test and development formed our initial training dataset.

Semi-automatic check. Even though very few in number, all resources (even expert-based ones) contain errors such as incorrect part-of-speech tags or ill-formed examples. Moreover, the extraction of examples and the mappings across resources were not always accurate. In order to have as few resource-specific and mapping errors as possible, all training, development and test sets were semi-automatically post-processed, either with small fixes whenever possible or by removing problematic instances otherwise.

2.1.2 Pruning

WordNet is known to be a fine-grained resource (Navigli, 2006). Often, different senses of the same word are hardly distinguishable from one another even for humans. For example, more than 40 senses are listed for the verb *run*, with many of them corresponding to similar concepts, e.g., “move fast”, “travel rapidly”, and “run with the ball”. In order to avoid this high-granularity, we performed an automatic pruning of the resource, removing instances with subtle sense distinctions. Sense clustering is not a very well-defined problem (McCarthy et al., 2016) and there are different strategies to perform this sense distinction (Snow et al., 2007; Pilehvar et al., 2013; Mancini et al., 2017). We adopted a simple strategy and

⁴Given that WordNet provides examples for synsets (rather than word senses), a target word (sense) might not occur in all the examples of its corresponding synset.

removed all pairs whose senses were first degree connections in the WordNet semantic graph, including sister senses, and those which belonged to the same supersense, i.e. sense clusters from the Wordnet lexicographer files⁵. There are a total of 44 supersenses in WordNet, comprising semantic categories such as *shape*, *substance* or *event*. This coarsening of the WordNet sense inventory has been shown particularly useful in downstream applications (Rüd et al., 2011; Severyn et al., 2013; Flekova and Gurevych, 2016; Pilehvar et al., 2017). In the next section we show that the pruning resulted in a significant boost in the clarity of the dataset.

2.2 Quality check

To verify the quality and the difficulty of the dataset and to estimate the human-level performance upperbound, we randomly sampled four sets of 100 instances from the test set, with an overlap of 50 instances between two of the annotators. Each set was assigned to an annotator who was asked to label each instance based on whether they thought the two occurrences of the word referred to the same meaning or not.⁶ The annotators were not provided with knowledge from any external lexical resource (such as WordNet). Specifically, the number of senses and the sense distinctions of the word (in the target sense inventory) were unknown to the annotators.

We found the average human accuracy on the dataset to be 80.0% (individual scores of 79%, 79%, 80% and 82%). We take this as an estimation of the human-level performance upperbound of the dataset. For the overlapping section, we computed the agreement between the two annotators to be 80%. Note that the annotators were not provided with sense distinctions to resemble the more difficult scenario for unsupervised models (which do not benefit from sense-based knowledge resources). Having access to sense definitions/distinctions would have substantially raised the performance bar.

Impact of pruning. To check the effectiveness of our pruning strategy, we also sampled a set of 100 instances from the batch of instances that were pruned from the dataset. Similarly, the annotators

⁵wordnet.princeton.edu/documentation/lexnames5wn

⁶Annotators were not lexicographers. To make the task more understandable, they were asked if in their opinion the two words would belong to the same dictionary entry or not.

| Split | Instances | Nouns | Verbs | Unique words |
|----------|-----------|-------|-------|--------------|
| Training | 5,428 | 49% | 51% | 1,256 |
| Dev | 638 | 62% | 38% | 599 |
| Test | 1,400 | 59% | 41% | 1,184 |

Table 2: Statistics of different splits of WiC.

were asked to independently label instances in the set. We computed the average accuracy on this set to be 57% (56% and 58%), which is substantially lower than that for the final pruned set (i.e. 80%). This indicates the success of our pruning strategy in improving the semantic clarity of the dataset.

2.3 Statistics

Table 2 shows the statistics of the different splits of WiC. The test set contains a large number of unique target words (1,256), reflecting the variety of the dataset. The large training split of 5,428 instances makes the dataset suitable for various supervised algorithms, including deep learning models. Only 36% of the target words in the test split overlap with those in the training, with no overlap of contextual sentences across the splits. This makes WiC extremely challenging for systems that heavily rely on pattern matching.

3 Experiments

We experimented with recent multi-prototype and contextualized word embedding techniques. Evaluation of other embedding models as well as word sense disambiguation systems is left for future work.

Contextualized word embeddings. One of the pioneering contextualized word embedding models is **Context2Vec** (Melamud et al., 2016), which computes the embedding for a word in context using a multi-layer perceptron which is built on top of a bidirectional LSTM (Hochreiter and Schmidhuber, 1997) language model. We used the 600-*d* UkWac pre-trained models⁷. **ELMo** (Peters et al., 2018) is a character-based model which learns dynamic word embeddings that can change depending on the context. ELMo embeddings are essentially the internal states of a deep LSTM-based language model, pre-trained on a large text corpus. We used the 1024-*d* pre-trained models⁸ for two configurations: ELMo₁, the first LSTM hidden state, and ELMo₃, the weighted sum of the

⁷<https://github.com/orenmel/context2vec>

⁸<https://www.tensorflow.org/hub/modules/google/elmo/1>

3 layers of LSTM. A more recent contextualized model is **BERT** (Devlin et al., 2019). The technique is built upon earlier contextual representations, including ELMo, but differs in the fact that, unlike those models which are mainly unidirectional, BERT is bidirectional, i.e., it considers contexts on both sides of the target word during representation. We experimented with two pre-trained BERT models: *base* (768 dimensions, 12 layer, 110M parameters) and *large* (1024 dimensions, 24 layer, 340M parameters).⁹ Around 22% of the pairs in the test set had at least one of their target words not covered by these models. For such out-of-vocabulary cases, we used BERT’s default tokenizer to split the unknown word to subwords and computed its embedding as the centroid of the corresponding subwords’ embeddings.

Multi-prototype embeddings. We experiment with three recent techniques that release 300-*d* pre-trained multi-prototype embeddings¹⁰. **JBT**¹¹ (Pelevina et al., 2016) induces different senses by clustering graphs constructed using word embeddings and computes embedding for each cluster (sense). **DeConf**¹² (Pilehvar and Collier, 2016) exploits the knowledge encoded in WordNet. For each sense, it extracts from the resource the set of semantically related words, called sense biasing words, which are in turn used to compute the sense embedding. **SW2V**¹³ (Mancini et al., 2017) is an extension of Word2Vec (Mikolov et al., 2013a) for jointly learning word and sense embeddings, producing a shared vector space of words and senses as a result. For these three methods we follow the disambiguation strategy suggested by Pelevina et al. (2016): for each example we retrieve the closest sense embedding to the context vector, which is computed by averaging its contained words’ embeddings.

Sentence-level baselines. We also report results for two baseline models which view the task as context (sentence) similarity. The **BoW** system views the sentence as a bag of words and computes a simple embedding as average of its words. The system makes use of Word2vec (Mikolov et al., 2013b) 300-*d* embeddings pre-trained

⁹<https://github.com/google-research/bert/blob/master/>

¹⁰*Multi-prototype* embeddings are also referred to as *sense* embeddings in the literature.

¹¹<https://github.com/uhh-lt/sensegram>

¹²<https://pilehvar.github.io/deconf/>

¹³<http://lcl.uniroma1.it/sw2v>

| | MLP | Threshold |
|---|-------------------|-------------|
| Contextualized word-based models | | |
| Context2vec | 57.9 ± 0.9 | 59.7 |
| EIMo ₁ | 56.4 ± 0.6 | 57.1 |
| EIMo ₃ | 57.2 ± 0.8 | 56.3 |
| BERT _{large} | 57.4 ± 1.0 | 63.8 |
| BERT _{base} | 60.2 ± 0.4 | 63.6 |
| Multi-prototype models | | |
| DeConf* | 52.4 ± 0.8 | 62.1 |
| SW2V* | 54.1 ± 0.5 | 59.1 |
| JBT | 54.1 ± 0.6 | 54.5 |
| Sentence-level baselines | | |
| BoW | 54.2 ± 1.3 | 61.0 |
| Sentence LSTM | 53.1 ± 0.9 | |

Table 3: Accuracy % performance of different models on the WiC dataset. The estimated (human-level) performance is 80.0 (cf. Section 2.2) and a random baseline would perform at 50.0. Systems marked with * make use of external lexical resources.

on the Google News corpus. **Sentence LSTM** is another baseline, which differently from the other models, does not obtain explicit encoded representations of the target word or sentence. The system has two LSTM layers with 50 units, one for each context side, which concatenates the outputs and passes that to a feedforward layer with 64 neurons, followed by a dropout layer at rate 0.5, and a final one-neuron output layer of sigmoid activation.

We used two simple binary classifiers in our experiments on top of all comparison systems (except for the LSTM baseline). **MLP**: a simple dense network with 100 hidden neurons (ReLU activation), and one output neuron (sigmoid activation), tuned on the development set (batch size: 32; optimizer: Adam; loss: binary crossentropy). Given the stochasticity of the network optimizer, we report average results for five runs (\pm standard deviation). **Threshold**: a simple threshold-based classifier based on the cosine distance of the two input vectors, tuned with step size 0.02 on the development set.

3.1 Results

Table 3 shows the results on WiC. In general, the dataset proves to be very difficult for all the techniques, with the best model, i.e., BERT_{large}, providing around 14% absolute improvement over a random baseline. Among the two classifiers, the simple threshold-based strategy, which computes

the cosine distance between the two encodings, proves to be more efficient than the MLP network which might not be suitable for this setting with small amount of training data. The 16.2% absolute accuracy difference between human-level upperbound and state-of-the-art performance suggests, however, a challenging dataset and encourages future research in context-sensitive word embeddings to leverage WiC in their evaluations.

Among the contextualized word-based models, after BERT, Context2vec provides more competitive results on the dataset. However, surprisingly, neither ELMo nor Context2vec are able to improve over the simple sentence BoW baseline (which also outperforms the sentence LSTM baseline) using the threshold strategy. This raises a question about the ability of these models in capturing fine-grained semantics of words in various contexts. Finally, as far as multi-prototype techniques are concerned, DeConf is the best performer. We note that DeConf indirectly benefits from sense-level information from WordNet encoded in its embeddings. The same applies to SW2V, which leverages knowledge from a significantly larger lexical resource, i.e., BabelNet (Navigli and Ponzetto, 2012).

4 Related work

The Stanford Contextual Word Similarity (SCWS) dataset (Huang et al., 2012) comprises 2003 word pairs and is analogous to standard word similarity datasets, such as RG-65 (Rubenstein and Goodenough, 1965) and SimLex (Hill et al., 2015), in which the task is to automatically estimate the semantic similarity of word pairs. Ideally, the estimated similarity scores should have high correlation with those given by human annotators. However, there is a fundamental difference between SCWS and other word similarity datasets: each word in SCWS is associated with a context which triggers a specific meaning of the word. The unique property of the dataset makes it a suitable benchmark for multi-prototype and contextualized word embeddings. However, in the following, we highlight some of the limitations of the dataset which hinder its suitability for evaluating existing techniques.

Inter-rater agreement (IRA) is widely accepted as a metric to assess the annotation quality of a dataset. The metric reflects the homogeneity of ratings which is expected to be high for a well-

defined task and a qualified set of annotators. For each word pair in SCWS ten scores were obtained through crowdsourcing. We computed the pairwise IRA to be 0.35 (in terms of Spearman ρ correlation) which is a very low figure. The mean IRA (between each annotator and the average of others), which can be taken as a human-level performance upperbound, is 0.52. Moreover, most of the instances in SCWS have context pairs with different target words.¹⁴ This makes it possible to test context-independent models, which only considers word pairs in isolation, on the dataset. Importantly, such a context-independent model can easily surpass the human-level performance upperbound. For instance, we computed the performance of the Google News Word2vec pre-trained word embeddings (Mikolov et al., 2013b) on the dataset to be 0.65 (ρ), which is significantly higher than the optimistic IRA for the dataset. In fact, Dubossarsky et al. (2018) showed how the reported high performance of multi-prototype techniques in this dataset was not due to an accurate sense representation, but rather to a subsampling effect, which had not been controlled for in similarity datasets. In contrast, a context-insensitive word embedding model would perform no better than a random baseline on our dataset.

5 Conclusions

In this paper we have presented a benchmark for evaluating context-sensitive word representations. The proposed dataset, WiC, is based on lexicographic examples, which constitute a reliable basis to validate different models in their ability to perceive and discern different meanings of words. We tested some of the recent state-of-the-art contextualized and multi-prototype embedding models on our dataset. The considerable gap between the performance of these models and the human-level upperbound suggests ample room for future work on modeling the semantics of words in context.

Acknowledgments

We would like to thank Luis Espinosa-Anke and Carla Pérez-Almendros for their help with the manual evaluation and Kiamehr Rezaee for running the BERT experiments.

¹⁴Only 8% (12% if ignoring PoS) of SCWS pairs are identical but their assigned scores (by average 6.8) are substantially higher than the dataset average of 3.6 on a [0,10] scale.

References

- Jose Camacho-Collados and Taher Pilehvar. 2018. From word to sense embeddings: A survey on vector representations of meaning. *Journal of Artificial Intelligence Research*, 63:743–788.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL*, Minneapolis, United States.
- Haim Dubossarsky, Eitan Grossman, and Daphna Weinshall. 2018. Coming to your senses: on controls and evaluation sets in polysemy research. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1732–1740, Brussels, Belgium.
- Christiane Fellbaum, editor. 1998. *WordNet: An Electronic Database*. MIT Press, Cambridge, MA.
- Lucie Flekova and Iryna Gurevych. 2016. Supersense embeddings: A unified model for supersense interpretation, prediction, and utilization. In *Proceedings of ACL*.
- Felix Hill, Roi Reichart, and Anna Korhonen. 2015. Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*.
- Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.
- Eric H. Huang, Richard Socher, Christopher D. Manning, and Andrew Y. Ng. 2012. Improving word representations via global context and multiple word prototypes. In *Proceedings of ACL*, pages 873–882, Jeju Island, Korea.
- Karin Kipper-Schuler. 2005. VerbNet: A broad-coverage, comprehensive verb lexicon. *Ph.D. thesis*.
- Massimiliano Mancini, Jose Camacho-Collados, Ignacio Iacobacci, and Roberto Navigli. 2017. Embedding words and senses together via joint knowledge-enhanced training. In *Proceedings of CoNLL*, pages 100–111, Vancouver, Canada.
- Diana McCarthy, Marianna Apidianaki, and Katrin Erk. 2016. Word sense clustering and clusterability. *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.
- Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781.
- Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.
- Roberto Navigli. 2006. Meaningful clustering of senses helps boost Word Sense Disambiguation performance. In *Proceedings of COLING-ACL*, pages 105–112, Sydney, Australia.
- 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.
- Arvind Neelakantan, Jeevan Shankar, Alexandre Passos, and Andrew McCallum. 2014. Efficient non-parametric estimation of multiple embeddings per word in vector space. In *Proceedings of EMNLP*, pages 1059–1069, Doha, Qatar.
- Maria Pelevina, Nikolay Arefyev, Chris Biemann, and Alexander Panchenko. 2016. Making sense of word embeddings. In *Proceedings of the 1st Workshop on Representation Learning for NLP*, pages 174–183.
- M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of NAACL*, New Orleans, LA, USA.
- Mohammad Taher Pilehvar, Jose Camacho-Collados, Roberto Navigli, and Nigel Collier. 2017. Towards a Seamless Integration of Word Senses into Downstream NLP Applications. In *Proceedings of ACL*, Vancouver, Canada.
- Mohammad Taher Pilehvar and Nigel Collier. 2016. De-conflated semantic representations. In *Proceedings of EMNLP*, pages 1680–1690, Austin, TX.
- Mohammad Taher Pilehvar, David Jurgens, and Roberto Navigli. 2013. Align, Disambiguate and Walk: a Unified Approach for Measuring Semantic Similarity. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, pages 1341–1351, Sofia, Bulgaria.
- Joseph Reisinger and Raymond J. Mooney. 2010. Multi-prototype vector-space models of word meaning. In *Proceedings of ACL*, pages 109–117.
- Herbert Rubenstein and John B. Goodenough. 1965. Contextual correlates of synonymy. *Communications of the ACM*, 8(10):627–633.
- Stefan Rüd, Massimiliano Ciaramita, Jens Müller, and Hinrich Schütze. 2011. Piggyback: Using search engines for robust cross-domain named entity recognition. In *Proceedings of ACL-HLT*, pages 965–975, Portland, Oregon, USA.

- Aliaksei Severyn, Massimo Nicosia, and Alessandro Moschitti. 2013. Learning semantic textual similarity with structural representations. In *Proceedings of ACL (2)*, pages 714–718, Sofia, Bulgaria.
- Rion Snow, Sushant Prakash, Daniel Jurafsky, and Andrew Y. Ng. 2007. Learning to merge word senses. In *Proceedings of EMNLP*, pages 1005–1014, Prague, Czech Republic.
- Yogarshi Vyas and Marine Carpuat. 2017. Detecting asymmetric semantic relations in context: A case-study on hypernymy detection. In *Proceedings of the 6th Joint Conference on Lexical and Computational Semantics (*SEM 2017)*, pages 33–43. Association for Computational Linguistics.
- George K. Zipf. 1949. *Human Behaviour and the Principle of Least-Effort*. Addison-Wesley, Cambridge, MA.