Not All Contexts Are Created Equal: Better Word Representations with Variable Attention

Wang Ling   Lin Chu-Cheng   Yulia Tsvetkov   Silvio Amir
Ramón Fernandez Astudillo   Chris Dyer   Alan W Black   Isabel Trancoso

L^{2}F Spoken Systems Lab, INESC-ID, Lisbon, Portugal
Language Technologies Institute, Carnegie Mellon University, Pittsburgh, PA, USA
Instituto Superior Técnico, Lisbon, Portugal
{lingwang,chuchenl,ytsvetko,cdyer,awb}@cs.cmu.edu
{ramon.astudillo,samir,isabel.trancoso}@inesc-id.pt

Abstract

We introduce an extension to the bag-of-words model for learning words representations that take into account both syntactic and semantic properties within language. This is done by employing an attention model that finds within the contextual words, the words that are relevant for each prediction. The general intuition of our model is that some words are only relevant for predicting local context (e.g. function words), while other words are more suited for determining global context, such as the topic of the document. Experiments performed on both semantically and syntactically oriented tasks show gains using our model over the existing bag of words model. Furthermore, compared to other more sophisticated models, our model scales better as we increase the size of the context of the model.

1 Introduction

Learning word representations using raw text data have been shown to improve many NLP tasks, such as part-of-speech tagging (Collobert et al., 2011), dependency parsing (Chen and Manning, 2014; Kong et al., 2014) and machine translation (Liu et al., 2014; Kalchbrenner and Blunsom, 2013; Devlin et al., 2014; Sutskever et al., 2014). These embeddings are generally learnt by defining an objective function, which predicts words conditioned on the context surrounding those words. Once trained, these can be used as features (Turian et al., 2010), as initializations of other neural networks (Hinton and Salakhutdinov, 2012; Erhan et al., 2010; Guo et al., 2014).

The continuous bag-of-words (Mikolov et al., 2013) is one of the many models that learns word representations from raw textual data. While these models are adequate for learning semantic features, one of the problems of this model is the lack of sensitivity for word order, which limits their ability of learn syntactically motivated embeddings (Ling et al., 2015a; Bansal et al., 2014). While models have been proposed to address this problem, the complexity of these models ("Structured skip-n-gram" and "CWindow") grows linearly as size of the window of words considered increases, as a new set of parameters is created for each relative position. On the other hand, the continuous bag-of-words model requires no additional parameters as it builds the context representation by summing over the embeddings in the window and its performance is an order of magnitude higher than of other models.

In this work, we propose an extension to the continuous bag-of-words model, which adds an attention model that considers contextual words differently depending on the word type and its relative position to the predicted word (distance to the left/right). The main intuition behind our model is that the prediction of a word is only dependent on certain words within the context. For instance, in the sentence *We won the game! Nicely played!*, the prediction of the word *played*, depends on both the syntactic relation from *nicely*, which narrows down the list of candidates to verbs, and on the semantic relation from *game*, which further narrows down the list of candidates to verbs related to games. On the other hand, the words *we* and *the* add very little to this particular prediction. On the other hand, the word *the* is important for predicting the word *game*, since it is generally followed by nouns. Thus, we observe that the same

word can be informative in some contexts and not in others. In this case, distance is a key factor, as the word the is informative to predict the immediate neighboring words, but not distance ones.

2 Attention-Based Continuous Bag-of-words

2.1 Continuous Bag-Of-Words (CBOW)

The work in (Mikolov et al., 2013) is frequently used to learn word embeddings. It defines projection matrix $\mathbf{W} \in \Re^{d \times |V|}$ where d is the embedding dimension with the vocabulary V. These parameters are optimized by by maximizing the likelihood that words are predicted from their context. Two models were defined, the skip-gram model and the continuous bag-of-words model. In this work, we focus on the continuous bag-of-words model. The CBOW model predicts the center word w_0 given a representation of the surrounding words $\mathbf{w}_{-b}, \ldots, \mathbf{w}_{-1}, \mathbf{w}_1, \mathbf{w}_b$, where b is a hyperparameter defining the window of context words. The context vector is obtained by averaging the embeddings of each word $\mathbf{c} = \frac{1}{2b} \sum_{i \in [-b,b]-\{0\}} \mathbf{w}_i$ and the prediction of the center word $\mathbf{w}_0$ is obtained by performing a softmax over all the vocabulary V. More formally, define the output matrix $\mathbf{O} \in \Re^{|V| \times d_w}$, which maps the context vector $\mathbf{c}$ into a $|V|$-dimensional vector representing the predicted word, and maximizes the following probability:

$$p(\mathbf{v}_0 \mid \mathbf{w}_{[-b,b]-\{0\}}) = \frac{\exp \mathbf{v}_0^\top \mathbf{O}\mathbf{c}}{\sum_{\mathbf{v} \in V} \exp \mathbf{v}^\top \mathbf{O}\mathbf{c}} \quad (1)$$

where $\mathbf{O}\mathbf{c}$ corresponds to the projection of the context vector $\mathbf{c}$ onto the vocabulary V and $\mathbf{v}$ is a one-hot representation. For larger vocabularies it is inefficient to compute the normalizer $\sum_{\mathbf{v} \in V} \exp \mathbf{v}^\top \mathbf{O}\mathbf{c}$. Solutions for problem are using the hierarchical softmax objective function (Mikolov et al., 2013) or resorting to negative sampling to approximate the normalizer (Goldberg and Levy, 2014).

The continuous bag-of-words model differs from other proposed models in the sense that its complexity does not rise substantially as we increase the window b, since it only requires two extra additions to compute $\mathbf{c}$, which correspond to d_w operations each. On the other hand, the skip-n-gram model requires two extra predictions corresponding to $d_w \times V$ operations each, which is an order of magnitude more expensive even when

subsampling V. However, the drawback the bag-of-words model is that it does not learn embeddings that are prone for learning syntactically oriented tasks, mainly due to lack of sensitivity to word order, since the context is defined by a sum of surrounding words. Extensions are proposed in (Ling et al., 2015a), where the sum if replaced by the concatenation of the word embeddings in the order these occur. However, this model does not scale well as b increases as it requires $V \times d_w$ more parameters for each new word in the window.

Finally, setting a good value for b is difficult as larger values may introduce a degenerative behavior in the model, as more effort is spent predicting words that are conditioned on unrelated words, while smaller values of b may lead to cases where the window size is not large enough include words that are semantically related. For syntactic tasks, it has been shown that increasing the window size can adversely impact in the quality of the embeddings (Bansal et al., 2014; Lin et al., 2015).

2.2 CBOW with Attention

We present a solution to these problems while maintaining the efficiency underlying the bag-of-words model, and allowing it to consider contextual words within the window in a non-uniform way. We first rewrite the context window c as:

$$\mathbf{c} = \sum_{i \in [-b,b]-\{0\}} a_i(w_i)\mathbf{w}_i \quad (2)$$

where we replace the average of the word embeddings with a weighted sum of the individual word embeddings within the context. That is, each word is w_i at relative position i is attributed an attention level representing how much the attention model believes this it is important to look at in order to predict the center word. The attention $a_i(w)$ given to word $w \in V$ at the relative position i is computed as:

$$a_i(w) = \frac{\exp k_{w,i} + s_i}{\sum_{j \in [-b,b]-\{0\}} \exp k_{w,j} + s_j} \quad (3)$$

where $\mathbf{K} \in \Re^{|V| \times 2b}$ (with elements $k_{i,j}$) is a set of parameters that which determines the importance of each word type in each (relative) position, $\mathbf{s} \in \Re^{2b}$ is a bias, which is conditioned only on the relative position. As this is essentially a softmax over context words, the default bag-of-words model can be seen as a special case of this model

where all parameters $\mathbf{K}$ and $\mathbf{s}$ are fixed at zero. Computing the attention of all words in the input requires $2b$ operations, as it simply requires retrieving one value from the lookup matrix $\mathbf{K}$ for each word and one value from the bias $\mathbf{s}$ for each word in the window. Considering that these models must be trainable on billions of tokens, efficiency is paramount. Although more sophisticated attentional models are certainly imaginable (Bahdanau et al., 2014), ours is a good balance of computational efficiency and modeling expressivity.

2.3 Parameter Learning

Gradients of the loss function with respect to the parameters $(\mathbf{W}, \mathbf{O}, \mathbf{K}, \mathbf{s})$ are computed with backpropagation, and parameters are updated after each training instance using a fixed learning rate.

3 Experiments

3.1 Word Vectors

We used a subsample from an English Wikipedia dump[1] containing 10 million documents, containing a total of 530 million tokens. We built word embeddings using the original CBOW and our proposed attentional model on this dataset.

In both cases, word vectors were constructed using window size $b = 20$, which enables us to capture longer-range dependencies between words. We set the embedding size $d_w = 50$ and used a negative sampling rate of 10. Finally, the vocabulary was reduced to words with more than 40 occurrences. In terms of computational speed, the original bag-of-words implementation was able to compute approximately 220k words per second, while our model computes approximately 100k words per second. The slowdown is tied to the fact that we are computing the gradients, the attention model parameters, as well as the word embeddings. On the other hand, the skip-n-gram model process words at only 10k words per second, as it must predict every word in the window b.

Figure 1 illustrates the attention model for the prediction of the word *south* in the sentence *antartica has little rainfall with the south pole making it a continental desert*. Darker cell indicate higher attention values from $a(i, w)$. We can observe that function words (has, the and a) tend to be attributed very low attentions, as these are generally less predictive power. On the other hand,
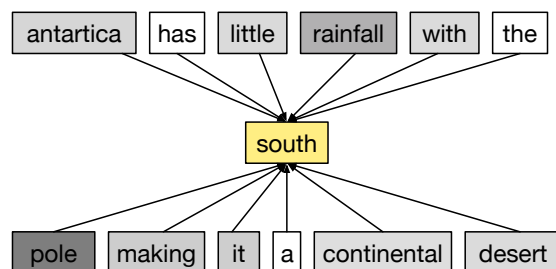
Figure 1: Illustration of the inferred attention parameters for a sentence from our training data when predicting the word *south*; darker cells indicate higher weights.

content words, such as antartica, rainfall, continental and desert are attributed higher weights as these words provide hints that the predicted word is likely to be related to these words. Finally, the word *pole* is assigned the highest attention as it close to the predicted word, and there is a very likely chance that *south* will precede *pole*.

3.2 Syntax Evaluation

For syntax, we evaluate our embeddings in the domain of part-of-speech tagging in both supervised (Ling et al., 2015b) and unsupervised tasks (Lin et al., 2015). This later task is newly proposed, but we argue that success in it is a compelling demonstration of separation of words into syntactically coherent clusters.

Part-of-speech induction. The work in (Lin et al., 2015) attempts to infer POS tags with a standard bigram hmm, which uses word embeddings to infer POS tags without supervision. We use the same dataset, obtained from the ConLL 2007 shared task (Nivre et al., 2007) Scoring is performed using the V-measure (Rosenberg and Hirschberg, 2007), which is used to predict syntactic classes at the word level. It has been shown in (Lin et al., 2015) that word embeddings learnt from structured skip-ngrams tend to work better at this task, mainly because it is less sensitive to larger window sizes. These results are consistent with our observations found in Table 1, in rows "Skip-ngram" and "SSkip-ngram". We can observe that our attention based CBOW model (row "CBOW Attention") improves over these results for both tasks and also the original CBOW model (row "CBOW").

	POS Induction	POS Tagging	Sentiment Analysis
CBOW	50.40	97.03	71.99
Skip-ngram	33.86	97.19	**72.10**
SSkip-ngram	47.64	**97.40**	69.96
CBOW Attention	**54.00**	97.39	71.39

Table 1: Results for unsupervised POS induction, supervised POS tagging and Sentiment Analysis (one per column) using different types of embeddings (one per row).

Part-of-speech tagging. The evaluation is performed on the English PTB, with the standard train (Sections 0-18), dev (Sections 19-21) and test (Sections 22-24) splits. The model is trained with the Bidirectional LSTM model presented in (Ling et al., 2015b) using the same hyper-parameters. Results on the POS accuracy on the test set are reported on Table 1. We can observe our model can obtain similar results compared to the structured skip-ngram model on this task, while training the model is significantly faster. The gap between the usage of different embeddings is not as large as in POS induction, as this is a supervised task, where pre-training generally leads to smaller improvements.

3.3 Semantic Evaluation

To evaluate the quality of our vectors in terms of semantics, we use the sentiment analysis task **(Senti)** (Socher et al., 2013), which is a binary classification task for movie reviews. We simply use the mean of the word vectors of words in a sentence, and use them as features in an ℓ_2-regularized logistic regression classifier. We use the standard training/dev/test split and report accuracy on the test set in table 1.

We can see that in this task, our models do not perform as well as the CBOW and Skipngram model, which hints that our model is learning embeddings that learn more towards syntax. This is expected as it is generally uncommon for embeddings to outperform existing models on both syntactic and semantic tasks simultaneously, as embeddings tend to be either more semantically or syntactically oriented. It is clear that the skipngram model learns embeddings that are more semantically oriented as it performs badly on all syntactic tasks. The structured skip-ngram model on the other hand performs badly on the syntactic tasks, but we observe a large drop on this semantically oriented task. Our attention-based model, on the other hand, out performs all other models on syntax-based tasks, while maintaining a competitive score on semantic tasks. This is an encouraging result that shows that it is possible to learn representations that can perform well on both semantic and syntactic tasks.

4 Related Work

Many methods have been proposed for learning word representations. Earlier work learns embeddings using a recurrent language model (Collobert et al., 2011), while several simpler and more lightweight adaptations have been proposed (Huang et al., 2012; Mikolov et al., 2013). While most of the learnt vectors are semantically oriented, work has been done in order to extend the model to learn syntactically oriented embeddings (Ling et al., 2015a). Attention models are common in vision related tasks (Tang et al., 2014), where models learn to pay attention to certain parts of a image in order to make accurate predictions. This idea has been recently introduced in many NLP tasks, such as machine translation (Bahdanau et al., 2014). In the area of word representation learning, no prior work that uses attention models exists to our knowledge.

5 Conclusions

In this work, we presented an extension to the CBOW model by introducing an attention model to select relevant words within the context to make more accurate predictions. As consequence, the model learns representations that are both syntactic and semantically motivated that do not degrade with large window sizes, compared to the original CBOW and skip-ngram models. Efficiency is maintained by learning a position-based attention model, which can compute the attention of surrounding words with a relatively small number of operations. Finally, we show improvements on syntactically oriented tasks, without degrading results significantly on semantically oriented tasks.

Acknowledgements

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473.

Mohit Bansal, Kevin Gimpel, and Karen Livescu. 2014. Tailoring continuous word representations for dependency parsing. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*.

Danqi Chen and Christopher D Manning. 2014. A fast and accurate dependency parser using neural networks. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 740–750.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *The Journal of Machine Learning Research*, 12:2493–2537.

Jacob Devlin, Rabih Zbib, Zhongqiang Huang, Thomas Lamar, Richard Schwartz, and John Makhoul. 2014. Fast and robust neural network joint models for statistical machine translation. In *52nd Annual Meeting of the Association for Computational Linguistics, Baltimore, MD, USA, June*.

Dumitru Erhan, Yoshua Bengio, Aaron Courville, Pierre-Antoine Manzagol, Pascal Vincent, and Samy Bengio. 2010. Why does unsupervised pre-training help deep learning? *The Journal of Machine Learning Research*, 11:625–660.

Yoav Goldberg and Omer Levy. 2014. word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method. *arXiv preprint arXiv:1402.3722*.

Jiang Guo, Wanxiang Che, Haifeng Wang, and Ting Liu. 2014. Revisiting embedding features for simple semi-supervised learning. In *Proceedings of EMNLP*.

Geoffrey E Hinton and Ruslan Salakhutdinov. 2012. A better way to pretrain deep boltzmann machines. In *Advances in Neural Information Processing Systems*, pages 2447–2455.

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 the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 873–882. Association for Computational Linguistics.

Nal Kalchbrenner and Phil Blunsom. 2013. Recurrent continuous translation models. In *EMNLP*, pages 1700–1709.

Lingpeng Kong, Nathan Schneider, Swabha Swayamdipta, Archna Bhatia, Chris Dyer, and Noah A. Smith. 2014. A dependency parser for tweets. In *Proc. of EMNLP*, pages 1001–1012, Doha, Qatar, October.

Chu-Cheng Lin, Waleed Ammar, Chris Dyer, and Lori Levin. 2015. Unsupervised POS induction with word embeddings. In *Proceedings of NAACL*.

Wang Ling, Chris Dyer, Alan Black, and Isabel Trancoso. 2015a. Two/too simple adaptations of word2vec for syntax problems. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics.

Wang Ling, Tiago Luíis, Luíis Marujo, Rámon Fernandez Astudillo, Silvio Amir, Chris Dyer, Alan W Black, and Isabel Trancoso. 2015b. Finding function in form: Compositional character models for open vocabulary word representation. EMNLP.

Shujie Liu, Nan Yang, Mu Li, and Ming Zhou. 2014. A recursive recurrent neural network for statistical machine translation. In *Proceedings of ACL*, pages 1491–1500.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, pages 3111–3119.

Joakim Nivre, Johan Hall, Sandra Kübler, Ryan McDonald, Jens Nilsson, Sebastian Riedel, and Deniz Yuret. 2007. The CoNLL 2007 shared task on dependency parsing. In *Proceedings of the CoNLL Shared Task Session of EMNLP-CoNLL 2007*, pages 915–932, Prague, Czech Republic, June. Association for Computational Linguistics.

Andrew Rosenberg and Julia Hirschberg. 2007. V-measure: A conditional entropy-based external cluster evaluation measure. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning(EMNLP-CoNLL)*, pages 410–420.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Y. Ng,

and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proc. of EMNLP*.

Ilya Sutskever, Oriol Vinyals, and Quoc VV Le. 2014. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems*, pages 3104–3112.

Yichuan Tang, Nitish Srivastava, and Ruslan R Salakhutdinov. 2014. Learning generative models with visual attention. In *Advances in Neural Information Processing Systems*, pages 1808–1816.

Joseph Turian, Lev Ratinov, and Yoshua Bengio. 2010. Word representations: A simple and general method for semi-supervised learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, pages 384–394, Stroudsburg, PA, USA. Association for Computational Linguistics.