Dependency-Enhanced Attention for Fact Extraction and Verification

Nimi Wang
Center for Data Science
New York University
nw1212@nyu.edu

Fangjun Zhang
Dept. of Computer Science
New York University
fz758@nyu.edu

Ruoyu Zhu
Center for Data Science
New York University
rz1403@nyu.edu

Abstract

The fact verification task has been proposed as an emerging research topic for ruling out misinformation on the Internet. Recently, Fact Extraction and VERification (FEVER) dataset was proposed as a benchmark for fact verification systems. FEVER task uses Wikipedia corpus as the database to verify claims using retrieved evidence. By analyzing existing work regarding this research topic, we found out that existing models sometimes fail to capture complex relationships among words in long sentences because of the absence of syntactic dependency. Thus, in this report, we propose dependency-enhanced self-attention, a variant of self-attention, aiming to mapping dependency relationships among words onto the attention layer. This module is also flexible to be used as an enhancement of self-attention in other tasks for better understanding of internal relationships in sentences.

Claim: Munich is the capital of Germany.
Retrieved Evidence:
[wiki/Germany]
Germany's capital and largest metropolis is Berlin, while its largest conurbation is the Ruhr (main centres: Dortmund and Essen).

[wiki/Munich]
Munich is the capital and largest city of the German state of Bavaria, on the banks of River Isar north of the Bavarian Alps.
Following a final reunification of the Wittelsbachian Duchy of Bavaria, previously divided and sub-divided for more than 200 years, the town became the country's sole capital in 1506.
Having evolved from a duchy's capital into that of an electorate (1623), and later a sovereign kingdom (1806), Munich has been a major European centre of arts, architecture, culture and science since the early 19th century, heavily sponsored by the Bavarian monarchs.

Label: Support

Figure 1: An example of negative output. The model is able to correctly select the evidence but fail to reveal the correct logical relation between the evidence and the claim. (Nie et al., 2019)

1 Introduction

The amounts of textual information sharing though the web has increased the demand for fact verification. In this context, the recent Fact Extraction and VERification (FEVER) shared task (Thorne et al., 2018) is launched to support the development of fact verification system.

The FEVER dataset we used in this paper consists of 185,445 claims manually verified against the provided dataset, which contains introductory sections of approximately 50,000 popular Wikipedia pages by June 2017. Each claim is labeled as SUPPORTED, REFUTED or NOTENOUGHINFO. These claims, generated by human annotators, need to focus on a single piece of information extracted from the all given Wikipedia pages. The annotators are allowed to mutate the raw texts about a single fact in different ways, so that the claims can be arbitrarily complicated, or to be paraphrases or negations of the original statements. For cases of SUPPORTED and REFUTED, the annotators also need to provide evidence from Wikipedia pages to either support or refute the claims. Accordingly, the goal of proposed fact verification systems need to not only output the correct label for a given claim, but also provide the corresponding evidence to justify the classification. A prediction is only considered as accurate if it gives the correct label for the claim and the extracted evidence set covers all ground truth evidence. The FEVER score is used to evaluate the accuracy of the prediction.

Concluding from proposed approaches by other FEVER competition participants, we found one of their common pitfalls is that these models fail to capture important yet complicated relationships among words in long sentences. One negative example is shown in Figure 1. Specifically, in this example the model does not capture the essential information ('state of Bavaria') at the end of sentence and is confused by the existence of lexical clue ('capital') in the evidence, resulting an incorrect prediction. Thus, we are motivated to apply dependency parsing in self-attention in our neural network model to improve this problem.

We provide a brief literature review in Section 2 first and describe details of our pipeline and proposed dependency-enhanced self attention in Section 3. We elaborate the settings for our experiments in Section 4 and the results in Section 5.

2 Related Work

This section presents a brief literature review for participants in FEVER competition. We will also include related works of natural language inference, attention-based model and dependency parsing.

Fact Extraction and Verification System is constructed by several different approaches. Nie et al. (2019) uses Neural Semantic Matching Network as their key component for semantic matching between two textual sequences. Another competition participant, the UCL Machine Reading Group (Yoneda et al., 2018), uses pre-trained Enhanced Sequential Inference Model (ESIM) (Chen et al., 2017) for natural language inference to predict the label of each pair of potential evidence sentences and the claim. UKP-Athene (Hanselowski et al., 2018) propose to use entity linking approach to find entities in given claims, in order to match the titles of Wikipedia articles.

Natural Language Inference (NLI), also known as Recognizing Textual Entailment (RTE), is a task which classifies the relationship between a pair of sentences as entailment, contradiction or neutral. With the increasing availability of large annotated data such as the Stanford Natural Language Inference (SNLI) (Bowman et al., 2015) and MultiNLI (Williams et al., 2018), training a complicated neural network is possible. Chen et al. (2017) further proposed an Enhanced Sequential Inference Model (ESIM) using inter-sentence attention between two LSTM layers. This model is recognized as one of the best-performing models and thus is used as the base of our neural network model in the Natural Language Inference module in this paper.

Attention-based model is initially used in machine translation (Bahdanau et al., 2014) and then widely used in natural language inference and reasoning (Parikh et al., 2016; Rocktäschel et al., 2015; Wang and Jiang, 2016). Self-attention (Cheng et al., 2016; Vaswani et al., 2017) is a context-aware variant of the original attention with the two input sentences being the same to capture intra-sentence relations. Self-attention mechanism is also widely used in other natural language processing tasks, such as machine comprehension (Cheng et al., 2016) and sentence embedding (Lin et al., 2017). Specifically, our self-attention model is flexible so that it can be used an enhanced feature to be applied to any attention-based models, such as BERT (Devlin et al., 2018).

Dependency parsing is to extract the intra-sentence information depending on the relationship between words. Many works involve deep learning approaches to solve the dependency parsing problems. A transition-based model combined with recurrent neural network incorporated the POS tags to predict the label for the dependency arc (Kiperwasser and Goldberg, 2016). Also, another neural network model based on LSTM (Hochreiter and Schmidhuber, 1997) is proposed with graph-based decoding for dependency types (Chen and Manning, 2014). Bi-affine attention mechanism is helpful to capture long-term information (Dozat and Manning, 2016). Instead of an end-to-end model, StanfordNLP (Qi et al., 2019) consists of four separate parts, tokenizer, POS tagger, lemmatizer and dependency parser. The dependency parser takes POS tagging information and tokens as input to a BiLSTM as well as a character-level convolutional neural network for character embedding. Our proposed module uses dependency parsing information as the input and the parser will not be updated.

3 Proposed Method

In this section, we present our baseline fact verification pipeline in detail. First we describe the overall pipeline used in our model and then we formally show the proposed Dependency-Enhanced Self-Attention module.

3.1 Baseline Model

Our baseline model pipeline extends the work from UCL Machine Reading Group (Yoneda et al., 2018) . The pipeline consists of four parts:

Document Retrieval module is responsible for selecting the related Wikipedia documents based on given claims. We first build up a dictionary of document titles and performs keyword matching by such titles to narrow down the set of selected documents. The probability of selecting the document is predicted by logistic regression using lexical features, including token-based matching amounts between the claim and first sentence in corresponding documents and presence of stop words.

Sentence Selection module retrieves sentences from chosen documents. Similar to document retrieval module, we uses a logistic regression model to retrieve most related sentences from the selected documents in the previous stage. Then the top ranked sentences are passed through the next Natural Language Inference stage.

Natural Language Inference uses the Enhanced Sequential Inference Model (ESIM) as the baseline model to label each pair of claim and retrieved evidence sentence. Both the claim sentence and the evidence sentence are encoded by a bidirectional LSTM (Hochreiter and Schmidhuber, 1997). The encoded sentence pairs are passed through a cross-aligned attention, which is similar to decomposable attention model (Parikh et al., 2016). For each pair, the model will predict the probabilities for SUPPORTED, REFUTED or NOTENOUGHINFO labels respectively. The ESIM model, which use bidirectional LSTM to encode claim and evidence sentences, is pre-trained on the Stanford Natural Language Inference corpus (Bowman et al., 2015) and then fine-tuned on FEVER dataset. We also use pre-trained ELMo (Peters et al., 2018) word embeddings.

In **Aggregation** module, we aggregate the three predicted scores and a evidence confidence score into the multi-layer perceptron (MLP) with 2 hidden layers. The MLP contains 100 hidden units for each layer and takes ReLU as the activation function.

3.2 Dependency-Enhanced Self-Attention

Let $\mathbf{a} = \{a_1, \dots, a_{l_a}\}$ be an input sentences of length l_a, and $a_i, \in \mathbf{R}^d$ where $i \in \{1, 2, ..., l_a\}$ is a word embedding vector of dimension d. Define a dependency mask matrix $D \in \mathbf{R}^{l_a \times l_a}$ of input sentence $\mathbf{a}$ where

$$D_{ij} = \begin{cases} c_p & \text{if dependency of type p} \\ 1 & \text{if no dependency} \end{cases}$$

where $c_p \in \mathbf{R}$ is a learned weight for the specific dependency relationship. Let E be the self attention weights matrix for sentence $\mathbf{a}$. We can obtain the unnormalized attention weights e_{ij} using function $F^{'}$ which is a RNN encoding model. The self-alignment between sentence $\mathbf{a}$ and itself can be decomposed as:

$$e_{ij} := F^{'}(a_i, a_j) = F(a_i)^{\top} F(a_j)$$

Then the dependency-enhanced soft alignment $\widehat{E}$ is defined as $\widehat{E} = D \odot E$ where $\odot$ indicates element-wise multiplication. Following the original attention mechanism, the attention weights is normalized as:

$$a_i' = \sum_{j=1}^{l_a} \frac{\exp(\hat{e}_{ij})}{\sum_{k=1}^{l_a} \exp(\hat{e}_{ik})} a_j$$

where $\hat{e}_{ij}$ is the entry at the i-th row and j-th column of matrix $\hat{E}$, standing for the dependency between i-th word and j-th word of the sentence. α_i' will have the same dimension with a_i and this reweighted embedding will be considered as input for attention construction in replacement of the original word embedding.

4 Experiments Setting

For dependency parsing, there are several packages available such as SpaCy[1], AllenNLP (Gardner et al., 2017) and StanfordNLP (Qi et al., 2019). We choose StanfordNLP library to preprocess all claims and retrieved evidence after sentence retrieval and before training the NLI model. For our NLI model, we use Jack the Reader (Weissenborn et al., 2018) as the implementation platform and rewrite codes for compatibility in PyTorch (Paszke et al., 2017).

Instead of using the whole FEVER dataset (180k+ claims), we randomly sample a subset of data (10k+) for training. Random down-sampling saves us time and prototypes our idea faster. Correspondingly, we down-sampled both the number of validation and testing dataset to 1k. As for training, we use Adam optimizer (Kingma and Ba, 2014) with a learning rate of 0.0004 and the decay of 0.9 each iteration. Batch size for training is 32. These values are chosen through empirical experiments. The code implementing our methods can be found in our GitHub repository.[2]

5 Results and Analysis

We evaluate the performance of our model quantitatively by the label accuracy and FEVER score.

[1] https://spacy.io/

[2] https://github.com/zfjmike/NLUProject

	Label Acc.	FEVER Score
ESIM	0.743	0.685
ESIM + SA	0.739	0.682
ESIM + Dep. SA	**0.744**	**0.689**

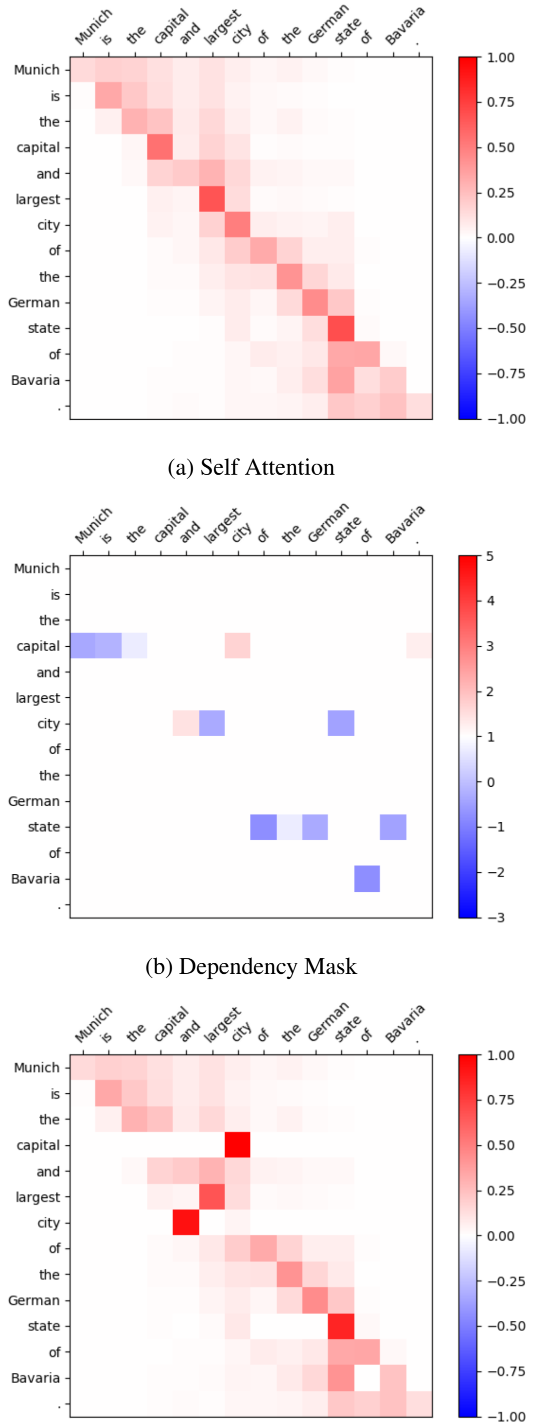
Table 1: Quantitative Results. **Dep.** stands for dependency-enhanced. **SA** stands for self-attention.

FEVER score is the accuracy when both the label is predicted right and the evidence set covers all ground truth evidence, while label accuracy focuses only on the label prediction. Results are reported in Table 1. We observe that baseline ESIM model with an additional self-attention layer, actually achieved both lower label accuracy and fever score, comparing with the baseline ESIM model (Chen et al., 2017). However, using our proposed dependency-enhanced self-attention layer instead of conventional self-attention, we achieve comparably higher label accuracy and fever score, comparing with the baseline.

Figure 2 illustrates the visualization of mentioned matrices. The first plot is the self-attention matrix of which the vertical and horizontal axis indicates the words of the sentence and the color shades at each cell represents the self-attention weights. The next plot is the dependency mask matrix, which is of the same size as self-attention matrix. The different color shades in each cell correspond to values of different dependency type for each word. The last plot is the resulted dependency-enhanced self-attention matrix which is computed by element-wise matrix multiplication of the previous two matrices. As we can see, the resulted dependency-enhanced self-attention matrix is different from the original self-attention matrix.

6 Conclusion

In this paper, we proposed the dependency-enhanced self-attention module, a variant of self-attention model. Specifically, we construct a learned dependency mask in which values represent different dependency types in a sentence. By applying this self-attention in natural language inference to predict label for the given claim and retrieved evidence, our model achieves a label accuracy of 0.744 and a FEVER score of 0.689. For future work, our proposed dependency-enhanced self-attention can be used as an enhanced feature for any attention-based models.



(a) Self Attention

(b) Dependency Mask

(c) Dependency-Enhanced Self-Attention

Figure 2: Illustration of result.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. *CoRR*, abs/1508.05326.

Danqi Chen and Christopher Manning. 2014. A fast and accurate dependency parser using neural networks. In *Proceedings of the 2014 conference on empirical methods in natural language processing*, pages 740–750.

Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Si Wei, Hui Jiang, and Diana Inkpen. 2017. Enhanced lstm for natural language inference. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1657–1668.

Jianpeng Cheng, Li Dong, and Mirella Lapata. 2016. Long short-term memory-networks for machine reading. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 551–561.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Timothy Dozat and Christopher D Manning. 2016. Deep biaffine attention for neural dependency parsing. *arXiv preprint arXiv:1611.01734*.

Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew Peters, Michael Schmitz, and Luke S. Zettlemoyer. 2017. Allennlp: A deep semantic natural language processing platform.

Andreas Hanselowski, Hao Zhang, Zile Li, Daniil Sorokin, Benjamin Schiller, Claudia Schulz, and Iryna Gurevych. 2018. Ukp-athene: Multi-sentence textual entailment for claim verification. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 103–108.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Eliyahu Kiperwasser and Yoav Goldberg. 2016. Simple and accurate dependency parsing using bidirectional lstm feature representations. *Transactions of the Association for Computational Linguistics*, 4:313–327.

Zhouhan Lin, Minwei Feng, Cicero Nogueira dos Santos, Mo Yu, Bing Xiang, Bowen Zhou, and Yoshua Bengio. 2017. A structured self-attentive sentence embedding. *arXiv preprint arXiv:1703.03130*.

Yixin Nie, Haonan Chen, and Mohit Bansal. 2019. Combining fact extraction and verification with neural semantic matching networks. In *Association for the Advancement of Artificial Intelligence (AAAI)*.

Ankur Parikh, Oscar Täckström, Dipanjan Das, and Jakob Uszkoreit. 2016. A decomposable attention model for natural language inference. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2249–2255.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. Automatic differentiation in pytorch. In *NIPS-W*.

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.

Peng Qi, Timothy Dozat, Yuhao Zhang, and Christopher D. Manning. 2019. Universal dependency parsing from scratch. *CoRR*, abs/1901.10457.

Tim Rocktäschel, Edward Grefenstette, Karl Moritz Hermann, Tomáš Kočiskỳ, and Phil Blunsom. 2015. Reasoning about entailment with neural attention. *arXiv preprint arXiv:1509.06664*.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. Fever: a large-scale dataset for fact extraction and verification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, volume 1, pages 809–819.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc.

Shuohang Wang and Jing Jiang. 2016. Learning natural language inference with lstm. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1442–1451.

D. Weissenborn, P. Minervine, T. Dettmers, I. Augenstein, J. Welbi, T. Rocktaschel, M. Bosnjak, J. Mitchell, Thomas Demeester, P. Stenetorp, and S. Riedel. 2018. Jack the reader : a machine reading framework. In *56th Annual Meeting of the Association for Computational Linguistics*, pages 1–7.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. 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 1112–1122. Association for Computational Linguistics.

Takuma Yoneda, Jeff Mitchell, Johannes Welbl, Pontus Stenetorp, and Sebastian Riedel. 2018. Ucl machine reading group: Four factor framework for fact finding (hexaf). In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 97–102.

Collaboration Statement

Fangjun Zhang, Nimi Wang and Ruoyu Zhu conceive the idea and design the model. Fangjun Zhang implements the model and performs the experiments on FEVER dataset. For writing part, Nimi Wang focuses on related work, proposed method, and results and analysis. Ruoyu Zhu focuses on baseline model. Fangjun Zhang focuses on details of the proposed model.