BERT for Evidence Retrieval and Claim Verification

Soleimani, A.; Monz, C.; Worring, M.

DOI
10.1007/978-3-030-45442-5_45

Publication date
2020

Document Version
Submitted manuscript

Published in
Advances in Information Retrieval

BERT for Evidence Retrieval and Claim Verification

Amir Soleimani, Christof Monz, Marcel Worring
Informatics Institute
University of Amsterdam
The Netherlands
{a.soleimani,c.monz,m.worring}@uva.nl

Abstract

Motivated by the promising performance of pre-trained language models, we investigate BERT in an evidence retrieval and claim verification pipeline for the FEVER fact extraction and verification challenge. To this end, we propose to use two BERT models, one for retrieving potential evidence sentences supporting or rejecting claims, and another for verifying claims based on the predicted evidence sets. To train the BERT retrieval system, we use pointwise and pairwise loss functions, and examine the effect of hard negative mining. A second BERT model is trained to classify the samples as supported, refuted, and not enough information. Our system achieves a new state of the art recall of 87.1 for retrieving top five sentences out of the FEVER documents consisting of 50K Wikipedia pages, and scores second in the official leaderboard with the FEVER score of 69.7.

1 Introduction

The constantly growing online textual information and the rise in the popularity of social media have been accompanied by the spread of fake news and false claims. It is not feasible to manually determine the truthfulness of such information. Therefore, there is a need for automatic verification and fact-checking. Due to the unavailability of proper datasets for evidence-based fake news detection, we focus on claim verification.

The Fact Extraction and VERification (FEVER) shared task (Thorne et al., 2018) introduces a benchmark for evidence-based claim verification. FEVER consists of 185K generated claims labelled as 'SUPPORTED', 'REFUTED' or 'NOT ENOUGH INFO' based on the introductory sections of a 50K popular Wikipedia pages dump. The benchmark task is to classify the veracity of textual claims and extract the correct evidence sentences required to support or refute the claims. The primary evaluation metric (FEVER score) is label accuracy conditioned on providing evidence sentence(s) unless the predicted label is 'NOT ENOUGH INFO', which does not need any specific evidence. Figure 1 shows three examples of the FEVER dataset.

Claim: Roman Atwood is a content creator. **Evidence:** **`[wiki/Roman_Atwood]`** He is best known for his vlogs, where he posts updates about his life on a daily basis. **Verdict:** SUPPORTED
Claim: Furia is adapted from a short story by Anna Politkovskaya. **Evidence:** **`[wiki/Furia_(film)]`** Furia is a 1999 French romantic drama film directed by Alexandre Aja, who co-wrote screenplay with Grgory Levasseur, adapted from the science fiction short story Graffiti by Julio Cortzar. **Verdict:** REFUTED
Claim: Afghanistan is the source of the Kushan dynasty. **Verdict:** NOT ENOUGH INFO

Figure 1: Three examples from the FEVER dataset (Thorne et al., 2018). Given a claim, the task is to to extract evidence sentence(s) from the Wikipedia dump and classify the claim as 'SUPPORTED', 'REFUTED', or 'NOT ENOUGH INFO'

To verify a claim, an enormous amount of information needs to be processed against the claim to retrieve the evidence and then infer possible

evidence-claim relations. This problem can be alleviated by a multi-step pipeline. Most work on FEVER has adopted a three-step pipeline (Figure 2): document retrieval, sentence retrieval, and claim verification. First, a set of documents, which possibly contain relevant information to support or reject a claim, are shortlisted from the Wikipedia dump. Second, five sentences are extracted out of the retrieved documents to be used as evidence. Third, the claim is verified against the retrieved evidence sentences.

The FEVER dataset covers a wide range of topics, and to overcome data limitation pre-trained models are useful. Recently the release of Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2018) has significantly advanced the performance in a wide variety of Natural Language Processing (NLP) tasks and datasets including MS MARCO (Nguyen et al., 2016) in passage re-ranking and MultiNLI (Williams et al., 2018) in natural language inference that respectively resemble the second and third step of the FEVER baseline. However, to the best of our knowledge, there is no integrated work for both steps.

In this paper, we propose a three-step pipeline system to address the FEVER task. We examine the BERT model for the FEVER task, and use that for evidence retrieval and claim verification. A first BERT model is trained to retrieve evidence sentences required for verifying the claims. We compare pointwise cross entropy loss and pairwise Hinge loss and Ranknet loss (Burges et al., 2005) for the BERT sentence retrieval. We further investigate the effect of Hard Negative Mining (HNM). Next, we train another BERT model to verify claims against the retrieved evidence sentences.

In summary, our contributions are as follows: (1) We employ the BERT model for evidence retrieval and claim verification; (2) We are the first to compare pointwise loss and pairwise loss functions for training the BERT model for sentence retrieval or fact extraction; (3) We investigate HNM to improve the retrieval performance; (4) We achieve second rank in the FEVER official leaderboard without ensembling.

2 Related Work

In this section, we first briefly survey related background in natural language inference. Second, we review particularly previous work on the FEVER task in the three-step pipeline: document retrieval, sentence retrieval, and claim verification

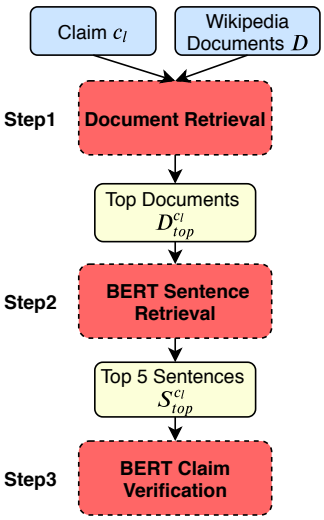

Figure 2: Three-step pipeline evidence extraction and claim verification.

2.1 Natural Language Inference

Natural Language Inference (NLI) is concerned with determining if a premise entails a hypothesis. The Stanford Natural Language Inference (SNLI) (Bowman et al., 2015) and the Multi-Genre NLI (MultiNLI) corpora (Williams et al., 2018) are the two established benchmarks for NLI. The availability of these large datasets has driven the recent advances in deep learning methods for NLI.

The deep models for NLI can be divided into two categories: (1) Models that classify the premise-hypothesis relation based on the concatenation of the premise and hypothesis fixed-size representations together with their element-wise products (Bowman et al., 2015; Bowman et al., 2016); (2) Uni-directional or bi-directional attention-based models that provide reasoning over distributional representation of the sentences (Rocktäschel et al., 2015; Chen et al., 2016).

In addition to the early improvement using context-free word representations (Mikolov et al., 2013; Pennington et al., 2014), pre-trained language models have significantly advanced several NLP tasks. In particular, BERT (Devlin et al., 2018) has achieved impressive results on several NLP tasks including NLI.

2.2 FEVER Pipeline

2.2.1 Document Retrieval

In the FEVER benchmark (Thorne et al., 2018), the DrQA (Chen et al., 2017) retrieval component is considered as the baseline. They choose the k-nearest documents based on the cosine similarity of TF-IDF feature vectors. In addition to the DrQA retrieval component, the Sweeper team (Hidey and Diab, 2018) considers lexical and syntactic features for the claim and first two sentences in the pages. The authors in (Malon, 2018) use TF-IDF along with exact matching of the page titles with the claim's named entities. The UCL team (Yoneda et al., 2018) highlights the pages titles, and detect them in the claims. They rank pages by logistic regression and extra features like capitalization, sentence position and token matching. Keyword matching along with page-view statistics are used in (Nie et al., 2019). UKP-Athene (Hanselowski et al., 2018), the highest document retrieval scoring team, uses MediaWiki API[1] to search the Wikipedia database for the claims noun phrases.

2.2.2 Sentence Retrieval

In order to extract evidence sentences, (Thorne et al., 2018) use a TF-IDF approach similar to their document retrieval. The UCL team (Yoneda et al., 2018) trains a logistic regression model on a heuristically set of features.

Enhanced Sequential Inference Model (ESIM) (Chen et al., 2016) with some small modifications has been used in (Nie et al., 2019; Hanselowski et al., 2018). ESIM encodes premises and hypotheses using one Bidirectional Long Short-Term Memory (BiLSTM) with shared weights. The encoded sentences are later aligned by a bidirectional attention mechanism. The encoded and aligned sentences are combined, and another shared BiLSTM matches the two representations. Finally, a softmax layer classifies the max and mean pooled representations of the second BiLSTM.

The UKP-Athene team (Hanselowski et al., 2018) achieved the highest sentence retrieval recall using ESIM and pairwise training. Their model takes a claim and a pair of positive and negative sentences and predicts a similarity score for each sentence. To train the model, they use a modified Hinge loss function and a random negative sampling strategy. In other words, positive samples are trained against five randomly selected negative sentences from the top retrieved pages for each claim. Note that recall is the most important factor in this step because the FEVER score counts a prediction as true if a complete set of evidence is retrieved.

2.2.3 Claim Verification

Decomposable Attention (DA) (Parikh et al., 2016), which compares and aggregates soft-aligned words in sentence pairs, is used in the FEVER benchmark paper (Thorne et al., 2018). The Papelo team (Malon, 2018) employs transformer networks with pre-trained weights (Radford et al., 2018). ESIM has been widely used among the FEVER challenge participants (Nie et al., 2019; Yoneda et al., 2018; Hanselowski et al., 2018). UNC (Nie et al., 2019), the winner of the competition, proposes a modified ESIM that takes the concatenation of the retrieved evidence sentences and claim along with ELMo embedding and three additional token-level features: WordNet, number embedding, and semantic relatedness score from the document retrieval and sentence retrieval steps. Dream (Zhong et al., 2019) has the state of the art FEVER score. The authors use a graph reasoning method based on XLNet (Yang et al., 2019) and RoBERTa (Liu et al., 2019), the two new BERT variants that are supposed to provide better pre-trained embeddings.

3 Methods

BERT (Devlin et al., 2018) is a multi-layer transformer language representation model pre-trained on the task of next sentence prediction and masked word prediction using extremely large datasets.

The input representation begins with a special classification embedding ([CLS]) followed by the tokens representations of the first and second sentences separated by another specific token ([SEP]).

In order to use the BERT model for different tasks, only one additional task-specific output layer is needed that can be trained together with fine-tuning the base layers. In particular, for the classification task, a softmax layer is added on the last hidden state of the first token, which is corresponding to [CLS]. Figure 3 demonstrates the BERT model components and structures. By default we use the BERT base model (12 layers) in all the experiments.

[1] https://www.mediawiki.org/wiki/API:Main_page

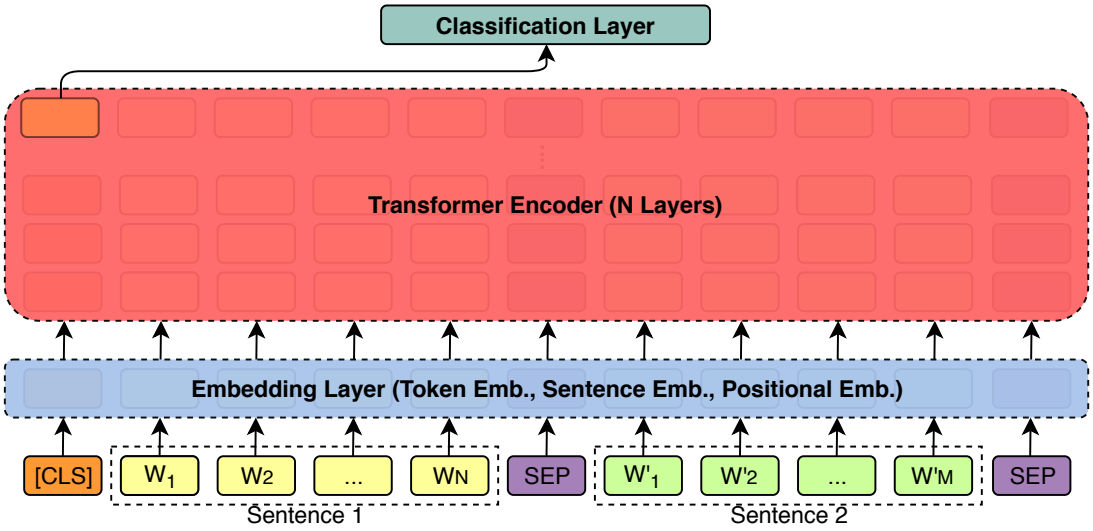

Figure 3: The BERT model takes the input with the form of [CLS] + Sentence 1 + [SEP] + Sentence 2 + [SEP], passes it through the embedding layer, which applies token, sentence, and positional embedding and N transformer encoder layers (BERT base:N=12, BERT large:N=24). Finally, a classification layer predicts the output from the first neuron of the last layer.

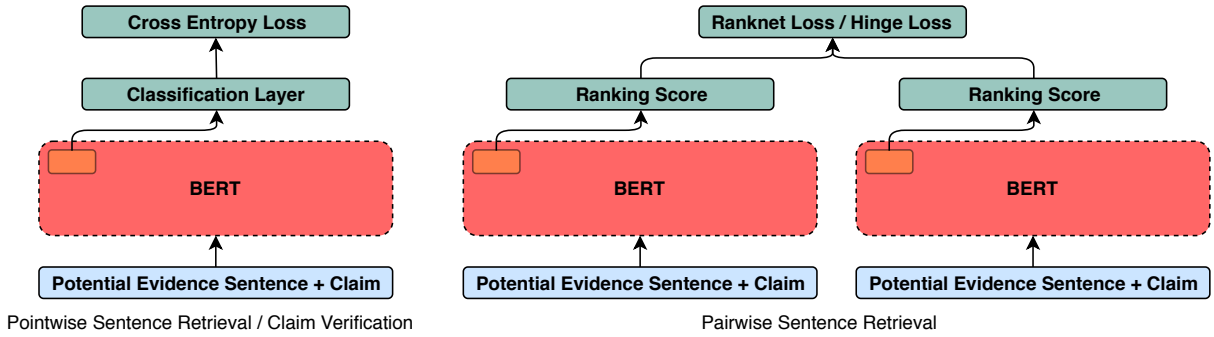

Figure 4: Pointwise sentence retrieval and claim verification (left), Pairwise sentence retrieval (right).

The FEVER dataset provides N_D Wikipedia documents $D = \{d_i\}_{i=1}^{N_D}$. The document d_i consists of sentences $S^{d_i} = \{s_j^i\}_{j=1}^{N_{S^{d_i}}}$. The goal is to classify the claim c_l for $l = 1, ..., N_C$ ($N_C = 145K$ for the FEVER benchmark) as 'SUPPORTED', 'REFUTED', or 'NOT ENOUGH INFO'. In order to count a prediction true, a complete set of evidence $E^{c_l} = \{s_j^i\}$ must be retrieved for the claim c_l. The claims with 'NOT ENOUGH INFO' label do not have an evidence set.

In this section, we explain the proposed system that we developed for the three FEVER steps. Figure 4 briefly demonstrates our proposed BERT-based architectures for the three-step pipeline (Figure 2).

3.1 Document Retrieval

In the document retrieval step, the Wikipedia documents containing the evidence supporting or refuting the claim are retrieved. Following the UKP-Athene promising document retrieval component (Hanselowski et al., 2018), which results in more than 93% development set document recall, we exactly use their method to collect a set of top documents $D_{top}^{c_l}$ for the claim c_l.

3.2 Sentence Retrieval

The sentence retrieval step extracts the top five potential evidence sentences $S_{top}^{c_l}$ for the claim c_l. The training set consists of about 145K claims and all the sentences (S^{d_i}) from the documents retrieved at the previous step ($D_{top}^{c_l}$) corresponding to the claim c_l ($S_{all}^{c_l} = \{S^{d_i} | d_i \in D_{top}^{c_l}\}$). Note

that $S^{c_l}_{all}$ may or may not contain the actual evidence sentences that we know from the ground-truth labels.

We adopt the pre-trained BERT model and fine-tune using two different pointwise and pairwise approaches. We did not observe any improvement to use the large BERT for this step. In both approaches, the input consist of a potential evidence sentence from $S^{c_l}_{all}$ and a claim c_l. Similar to (Malon, 2018), in order to compensate for the missed co-reference pronouns in the sentences, we add the Wikipedia pages titles at the beginning of each potential sentence. For all the retrieval experiments, we adopt a batch size of 32, a learning rate of $2e-5$, and one epoch of training.

3.2.1 Pointwise

In the pointwise approach, every single input is classified as evidence or non-evidence. We use cross entropy classification loss for the pointwise approach:

$$Loss_{point} = -\sum_{i=1}^{N} y_i \log(p_i) \quad (1)$$

where y_i and p_i are respectively the one-hot ground-truth label vector and the corresponding softmax output (Figure 4 (left)), and N is the total number of training samples.

At testing time, sentences are sorted by their p_i values and the top five sentences are considered as evidence. A threshold can also be used on the output scores to filter out uncertain results and trade-off the recall against the precision.

3.2.2 Pairwise

In the pairwise approach, a pair of positive and negative samples are compared against each other (Figure 4 (right)). We use the Ranknet loss function (Burges et al., 2005):

$$Loss^{Ranknet}_{pair} = -\sum_{i=1}^{N} \log p'_i \quad (2)$$

where the mapping from the positive sample o_{pos} and negative sample output o_{neg} to probabilities are calculated using the softmax function $p'_i = e^{o_{pos}-o_{neg}}/(1+e^{o_{pos}-o_{neg}})$. Note that we do not force the positive and negative samples to be selected from the same claims because the number of sentences per claim is significantly different and this difference might result in biasing on the claims with higher number of sentences.

In addition, we experiment with the modified Hinge loss functions like (Hanselowski et al., 2018):

$$Loss^{Hinge}_{pair} = \sum_{i=1}^{N} \max(0, 1 + o_{neg} - o_{pos}) \quad (3)$$

At testing time, for both pairwise loss functions, we sort the sentences by their output value o and similarly choose $S^{c_l}_{top}$ for the claim c_l.

3.2.3 Hard Negative Mining

The ratio of negative (non-evidence) to positive (evidence) sentences is high, thus it is not reasonable to train on all the negative samples. Random sampling limits the number of negative samples, however, this might lead to training on easy and trivial samples. Therefore, we opt to investigate the effect of HNM.

Similar to (Schroff et al., 2015), we focus on online HNM. We fix the positive samples batch size of 16 but heuristically increase negative sample batch size from 16 to 64 and train on the positive samples and only the 16 negative samples with the highest loss values. This results in a balanced batch sized of 32. In the case of pairwise retrieval, HNM is applied to select the 32 hardest pairs out of 128 pairs thus plenty of the positive samples might not be trained on. Therefore, for this case, we heuristically increase the training epoch from one to three. While increasing the epoch for the HNM improves the performance, we observed the reverse for the normal training. In the experiments without HNM, we use random negative sampling. Note that for both cases, loss values are computed in the no-gradient mode, like the inference time, and thus there is no need for more GPUs than normal training with the batch size of 32.

3.3 Claim Verification

In the final step, the top five potential evidence sentences $S^{c_l}_{top}$ are independently compared against the claim c_l and the final label is determined by aggregating the five individual decisions. Like (Malon, 2018), the default label is 'NOT ENOUGH INFO' unless there is any supporting evidence to predict the claim label as 'SUPPORTED'. If there is at least one piece of evidence rejecting the claim while there is no supporting fact, the final decision is 'REFUTED'.

We propose to train a new pre-trained BERT model as a three-class classifier (Figure 4 (left)).

Model	Precision(%)	Recall@5(%)	F1(%)
UNC (Nie et al., 2019)	36.39	86.79	51.38
UCL (Yoneda et al., 2018)	22.74**	84.54	35.84
UKP-Athene (Hanselowski et al., 2018)	23.67*	85.81*	37.11*
DREAM-XLNet (Zhong et al., 2019)	26.60	87.33	40.79
DREAM-RoBERTa (Zhong et al., 2019)	26.67	87.64	40.90
Pointwise	25.14	88.25	39.13
Pointwise + Threshold	**38.18**	88.00	**53.25**
Pointwise + HNM	25.13	88.29	39.13
Pairwise Ranknet	24.97	88.20	38.93
Pairwise Ranknet + HNM	24.97	**88.32**	38.93
Pairwise Hinge	24.94	88.07	38.88
Pairwise Hinge + HNM	25.01	88.28	38.98

Table 1: Development set sentence retrieval performance. * We calculated the scores using the official code, and for ** we used the F1 formula to calculate the score.

Model	FEVER Score(%)	Label Accuracy(%)
UNC (Nie et al., 2019)	66.14	69.60
UCL (Yoneda et al., 2018)	65.41	69.66
UKP-Athene (Hanselowski et al., 2018)	64.74	-
BERT & UKP-Athene	69.79	71.70
BERT Large & UKP-Athene	70.64	72.72
BERT & BERT (Pointwise)	71.38	73.51
BERT & BERT (Pointwise + HNM)	71.33	73.54
BERT (Large) & BERT (Pointwise)	**72.42**	74.58
BERT (Large) & BERT (Pointwise + HNM)	**72.42**	**74.59**
BERT & BERT (Pairwise Ranknet)	71.02	73.22
BERT & BERT (Pairwise Ranknet + HNM)	70.99	73.02
BERT & BERT (Pairwise Hinge)	71.60	72.74
BERT & BERT (Pairwise Hinge + HNM)	70.70	72.76

Table 2: Development set verification scores.

We train the model on $722K$ evidence-claim pairs provided by the first two steps. We adopt the batch size of 32, the learning rate of $2e-5$, and two epochs of training.

4 Results

Table 1 compares the development set performance of different variants of the proposed sentence retrieval method with the state of the art results on the FEVER dataset. The results indicate that both pointwise and pairwise BERT sentence retrieval improve the recall. The UNC and DREAM precision scores are better than our methods without a decision threshold, however, a threshold can regulate the trade-off between the recall and precision, and achieve the best precision and F1 scores. As discussed in (Nie et al., 2019), the recall is the most important factor. It is because the sentence retrieval predictions are the samples that we train the verification system on. Moreover, the FEVER score requires evidence for 'SUPPORTED' and 'REFUTED' claims. Therefore, we opt to focus more on recall and train the claim verification model on the predictions with the maximum recall. Surprisingly, the DREAM paper (Zhong et al., 2019) reports lower recalls for RoBERTa and XLNet that might be because of different training setups.

Although the pairwise Ranknet with HNM has the best recall score, we cannot conclude that pairwise methods are necessarily better for this task. This is more clear in Figure 5, which plots the recall-precision trade-off by applying a decision threshold on the output scores. The pointwise

Model	FEVER Score(%)	Label Accuracy(%)
DREAM (Zhong et al., 2019)	**70.60**	**76.85**
BERT (Large) & BERT (Pointwise + HNM)	69.66	71.86
abcd_zh*	69.40	72.81
BERT (Large) & BERT (Pointwise)	69.35	71.48
cunlp*	68.80	72.47
BERT & BERT (Pointwise)	68.50	70.67
BERT (Large) & UKP-Athene	68.36	70.41
BERT & FEVER UKP-Athene	67.49	69.40
UNC (Nie et al., 2019)	64.21	68.21
UCL (Yoneda et al., 2018)	62.52	67.62
UKP-Athene (Hanselowski et al., 2018)	61.58	65.46

Table 3: Results on the test set as of the date of writing (September 2019). * Unpublished methods listed on the leaderboard on codalab.

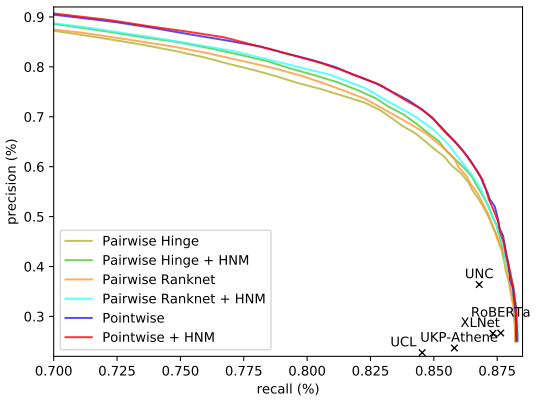

Figure 5: Recall and precision results on the development set. x shows the UNC, UCL, UPK-Athene, DREAM XLNet, and DREAM RoBERTa scores (Nie et al., 2019; Yoneda et al., 2018; Hanselowski et al., 2018; Zhong et al., 2019)

methods surpass the pairwise methods in terms of recall-precision performance. Figure 5 also shows that HNM enhances both pairwise methods trained by the Ranknet and Hinge loss functions and preserves the pointwise performance.

In Table 2, we compare the development set results of the state of the art methods with the BERT model trained on different retrieved evidence sets. The BERT claim verification system even if it is trained on the UKP-Athene sentence retrieval component (Hanselowski et al., 2018), the state of the art method with the highest recall, improves both label accuracy and FEVER score. Training based on the BERT sentence retrieval predictions significantly enhances the verification results because while it explicitly improves the FEVER score by providing more correct evidence sentences, it provides a better training set for the verification system. The large BERTs are only trained on the best retrieval systems, and as expected significantly improve the performance.

Finally, we report the blind test set results in Table 3 using the official FEVER framework on CodaLab[2] as of the date of writing. Our best model ranks at the second place that indicates the importance of using pre-trained language modelling methods for both sentence retrieval and claim verification systems. Note that it is not completely fair to compare our method with the DREAM's core idea because in addition to a graph-based reasoning approach they use XLNet, a superior pre-trained language model.

5 Conclusion

We investigated the BERT model for evidence sentence retrieval and claim verification. In the retrieval step, we compared the pointwise and pairwise approaches and concluded that although the pairwise Ranknet approach achieved the highest recall, pairwise approaches are not necessarily superior to the pointwise approach particularly if precision is taken into account. Our large system scored second with a FEVER score of 69.66 without ensembling.

We additionally examined hard negative mining for training the retrieval systems and showed that it slightly improves the performance. We discussed

[2] https://competitions.codalab.org/competitions/18814#results

that by constantly switching between the training and inference mode, the online hard negative mining does not require additional GPUs. We leave its probable effect on the faster training to future work. Furthermore, using BERT as an end-to-end framework for the entire FEVER pipeline can be investigated in the future.

Acknowledgments

This research was partly supported by VIVAT.

References

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.

Samuel R. Bowman, Jon Gauthier, Abhinav Rastogi, Raghav Gupta, Christopher D. Manning, and Christopher Potts. 2016. A fast unified model for parsing and sentence understanding. *CoRR*, abs/1603.06021.

Christopher Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Gregory N Hullender. 2005. Learning to rank using gradient descent. In *Proceedings of the 22nd International Conference on Machine learning (ICML-05)*, pages 89–96.

Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Si Wei, and Hui Jiang. 2016. Enhancing and combining sequential and tree LSTM for natural language inference. *CoRR*, abs/1609.06038.

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading wikipedia to answer open-domain questions. *CoRR*, abs/1704.00051.

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.

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, Brussels, Belgium, November. Association for Computational Linguistics.

Christopher Hidey and Mona Diab. 2018. Team SWEEPer: Joint sentence extraction and fact checking with pointer networks. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 150–155, Brussels, Belgium, November. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Christopher Malon. 2018. Team papelo: Transformer networks at FEVER. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 109–113, Brussels, Belgium, November. Association for Computational Linguistics.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 3111–3119. Curran Associates, Inc.

Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. MS MARCO: A human generated machine reading comprehension dataset. *CoRR*, abs/1611.09268.

Yixin Nie, Haonan Chen, and Mohit Bansal. 2019. Combining fact extraction and verification with neural semantic matching networks. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):6859–6866, Jul.

Ankur P. Parikh, Oscar Täckström, Dipanjan Das, and Jakob Uszkoreit. 2016. A decomposable attention model for natural language inference. *CoRR*, abs/1606.01933.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar, October. Association for Computational Linguistics.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training. *Technical report, OpenAI*.

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

Florian Schroff, Dmitry Kalenichenko, and James Philbin. 2015. Facenet: A unified embedding for face recognition and clustering. *CoRR*, abs/1503.03832.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. FEVER: a large-scale dataset for fact extraction and verification. *CoRR*, abs/1803.05355.

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.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *arXiv preprint arXiv:1906.08237*.

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, Brussels, Belgium, November. Association for Computational Linguistics.

Wanjun Zhong, Jingjing Xu, Duyu Tang, Zenan Xu, Nan Duan, Ming Zhou, Jiahai Wang, and Jian Yin. 2019. Reasoning over semantic-level graph for fact checking. *arXiv preprint arXiv:1909.03745*.