Improving Large-Scale Fact-Checking using Decomposable Attention Models and Lexical Tagging

Nayeon Lee*, **Chien-Sheng Wu***, **Pascale Fung**
Human Language Technology Center (HLTC)
Center for Artificial Intelligence Research (CAiRE)
Hong Kong University of Science and Technology
[nyleeaa,cwuak].connect.ust.hk, pascale@ece.ust.hk

Abstract

Fact-checking of textual sources needs to effectively extract relevant information from large knowledge bases. In this paper, we extend an existing pipeline approach to better tackle this problem. We propose a neural ranker using a decomposable attention model that dynamically selects sentences to achieve promising improvement in evidence retrieval F1 by 38.80%, with (×65) speedup compared to a TF-IDF method. Moreover, we incorporate lexical tagging methods into our pipeline framework to simplify the tasks and render the model more generalizable. As a result, our framework achieves promising performance on a large-scale fact extraction and verification dataset with speedup.

1 Introduction

With the rapid growth of available textual information, automatic extraction and verification, also known as fact-checking, has become important in order to identify relevant and factual information from the ever-growing information pool. The FakeNews Challenge (Pomerleau and Rao) addresses fact-checking as a simple stance detection problem, where the article is verified by checking the stance agreement between an article's title and content. Similar to the FakeNews, (Rashkin et al., 2017; Vlachos and Riedel, 2014) focused on political statements from Politifact.com to verify the degree of truthfulness. However, they assume that the gold standard documents containing the evidence are already known, which overly simplifies the task.

Question Answering (QA) is similar to fact-checking in the sense that a question and its answers can be considered as a claim and evidence respectively, but the answers may come from a large-scale database. Several approaches (Chen et al., 2017a; Ryu et al., 2014; Ahn et al., 2004) proposed QA system utilizing resources such as Wikipedia, which is more comprehensive and incorporates wider world knowledge. However, the main focus is to identify only the "correct" answers that support a given question. Since the ability to refute is as important as to support, it does not fully address the verification problem of fact-checking.

Claim	Finding Dory was written by anyone but an American.
Evidence	**Finding_Dory**: Directed by Andrew Stanton with co-direction by Angus MacLane, the screenplay was *written by Stanton and Victoria Strouse* **Andrew_Stanton**: Andrew Stanton -LRB- born December 3, 1965 -RRB- is an *American film director* , screenwriter, producer and voice actor based at Pixar.
Label	REFUTE

Table 1: Example of verified claim with evidence from multiple Wikipedia pages

Recently, Thorne et al. (2018) proposed a public dataset to explore the complete process of the large-scale fact-checking. It is designed not only to verify claims but also to extract sets of related evidence. Nevertheless, the pipeline solution proposed in that paper suffers from following problems: 1) The overall performance (30.88% accuracy) still needs further improvement to be applicable to the evidence selection and classification, which also highlights the challenging nature of this task. 2) The evidence retrieval using Term Frequency-Inverse Document Frequency (TF-IDF) is time-consuming since the TF-IDF between a claim and set of candidate evidence cannot be computed in advance.

In this paper, we extend the original pipeline solution to achieve faster and better fact-checking results. Our main contributions are: 1) Propose a neural ranker using decomposable attention (DA) model for evidence selection to speed up (×65) and outperform related works. 2) Incorporate several lexical tag information to effectively simplify

* These two authors contributed equally.

Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 1133–1138
Brussels, Belgium, October 31 - November 4, 2018.

k	DR_{tfidf}	DR_{rerank}
1	0.3145	0.6099
2	0.4321	0.7292
5	0.5895	0.8052
10	0.6916	0.8322
25	0.7882	0.8494
100	0.8886	0.8886

Table 2: Oracle document retrieval macro-recall in the test set (SUPPORT/REFUTE).

the problem and generalize the models. 3) Improve the overall fact extraction F1 by 38.80% and verification accuracy by 2.10% to achieve the state-of-the-art performance on the dataset.

2 Methodology

Our pipeline framework[1] has three main modules: document retrieval (DR), evidence selection (ES), and textual entailment recognition (TER). The goal is to verify a given claim with a set of evidence from Wikipedia (Table 1). The verification labels are support, refute and not enough information (NEI) to verify.

2.1 Lexical Tagging

In our framework, two lexical tags (i.e. part-of-speech (POS) and named entity recognition (NER)) are used to enhance the performance. We compute the tags for claims in advance using the Stanford CoreNLP (Manning et al., 2014) library. Using this information is helpful in the following ways: 1) it helps keyword extraction for each claim. 2) it reduces the out-of-vocabulary (OOV) problems related to name or organization entities, for better generalization. For example, a claim like "Michael Jackson and Justin Timberlake are friends," is replaced as "PERSON-1 and PERSON-2 are friends". In this way, we encourage our model to learn verification without dealing with the real entity values but the delexicalized indexed tokens.

2.2 Document Retrieval (DR)

For document retrieval, we extend the method of DrQA (Chen et al., 2017a), which calculates cosine similarity between query and document, using binned unigram and bigram TF-IDF features. We refer to this method as DR_{tfidf}.

Instead of directly selecting top k document using TF-IDF as in DR_{tfidf}, our document retriever DR_{rerank} use TD-IDF to reduce the search space from 5.4M to 100 documents. Re-ranking is then applied to select the top k documents. For re-rank, we defined a score function f_{rank} that ranks the relevance of the document by considering both the title and the content as follows:

$$r_{claim} = \frac{\text{POS}_{match}}{\text{POS}_{claim}}, r_{title} = \frac{\text{POS}_{match}}{\text{POS}_{title}},$$
$$f_{rank} = r_{claim} \times r_{title} \times tf\text{-}idf$$

To capture the relevance from the title, all the POS tags with high discriminating power (NN, NNS, NNP, NNPS, JJ, CD) of a claim are chosen as keywords. POS_{claim} and POS_{title} are the counts of such POS tags inside the claim and title respectively. POS_{match} is the count of common POS keywords in the claim and the title; r_{claim} is a ratio between POS_{match} and POS_{claim} to reward the documents with higher keyword hits; r_{title} is the ratio between POS_{match} and POS_{title} to penalize those documents with more candidate keywords as it is more likely to have keyword hits with more candidates. We incorporate the TF-IDF score ($tf\text{-}idf$) to ensure that the content information is not neglected. Our experiments show that our re-rank strategy increases the document recall compared to the single-step approach (Table 2). To decide on the optimal value for hyper-parameter k, full-pipeline performance was compared to evaluate the effect of k on final verification accuracy.

2.3 Evidence Selection (ES)

In this module, l sentences are extracted as possible evidence for the claim. Instead of selecting the sentences by recomputing sentence-level TF-IDF features between claim and document text as in Thorne et al. (2018), we propose a neural ranker using decomposable attention (DA) model (Parikh et al., 2016) to perform evidence selection. DA model does not require the input text to be parsed syntactically, nor is an ensemble, and it is faster without any recurrent structure. In general, using neural methods is better for the following reasons: 1) The TF-IDF may have limited ability to capture semantics compared to word representation learning 2) Faster inference time compared to TF-IDF methods that need real-time reconstruction.

The neural ranker DA_{rank} is trained using a fake task, which is to classify whether a given sentence is an evidence of a given claim or not. The

[1] https://github.com/HLTCHKUST/fact-checking

		TF-IDF	DA_{rank}			DA_{rank}+NER		
			1:1	1:4	1:9	1:1	1:4	1:9
l	2	0.847	0.170	0.889	0.889	0.109	0.889	**0.893**
	5	0.918	0.451	0.966	**0.968**	0.345	0.962	**0.968**
Time		3.57s	0.055s					

Table 3: Oracle evidence selection macro-recall in the test set using gold documents (SUPPORT/REFUTE).

output of DA_{rank} is considered as the evidence probability. The training samples are unbalanced since there are more unrelated sentences than evidence sentences. Note that the classifier's accuracy on the fake task is not crucial because the choice of evidence is based on the relative score of relevance compared to other candidates. Therefore, it is not necessary to balance positive and negative samples using up/down-sampling, to the contrary, making it unbalanced actually improved the performance (Table 3).

Unlike the k value which is fixed, the l value is selected dynamically based on the evidence score of DA_{rank}'. It is used as a confidence measure of the given sentence being an evidence. Evidence with the score below fixed threshold value th is eliminated. Hence, each claim will have a different number of l evidence. To decide on th, we also carry out the full-pipeline evaluation. We propose the dynamic selection of l because we hypothesize that any wrong evidence, or noise, from early module could harm the subsequent RTE module.

2.4 Recognizing Textual Entailment (RTE)

Given a claim and l possible evidence, a DA_{rte} classifier is trained to recognize the textual entailment to be support, refute or not enough information to verify (NEI). Same as Thorne et al. (2018), we use the decomposable attention (DA) between the claim and the evidence for RTE. DA model decomposes the RTE problem into subproblems, which can be considered as bi-direction word-level attention features. Note that the DA model is utilized over other models such as as Chen et al. (2017b); Glockner et al. (2018), because it is a simple but effective model.

Our DA_{rte} model must correctly decide whether a claim is NEI, when the evidence retrieved is irrelevant and insufficient. However, NEI claims have no annotated evidence, thus cannot be used to train RTE. To overcome this issue, same as (Thorne et al., 2018), the most probable NEI evidence are simulated by sampling sentences from the nearest page to the claim using the document retrieval module.

	MLP	DA_{rte}	DA_{rte}+NER
Accuracy (%)	63.2	78.4	**79.9**

Table 4: Oracle RTE classification accuracy in the test set using gold evidence.

3 Experimental setup

Dataset: FEVER dataset (Thorne et al., 2018) is a relatively large-scale dataset compared to other previous fact extraction and verification works, with around 5.4M Wikipedia documents and 185k samples. The claims are generated by altering sentences extracted from Wikipedia, with human-annotated evidence sentences and verification labels (e.g. Table 1). The training/validation/test sets of these three datasets are split in advance by the providers. Note that the test-set was equally split into 3 classes: Supported (3333), Refuted (3333), NEI (3333).

Training: We trained our models end-to-end using Adagrad optimizer (Duchi et al., 2011). The embedding size is set to 200 and initialized with GloVe (Pennington et al., 2014). The dropout rate is set to 0.2. In all the datasets, we tuned the hyper-parameters with grid-search over the validation set.

Evaluation: For each module, we independently measure oracle performance, where we assume gold standard documents and set of evidence are provided (oracle evaluation). For the final full-pipeline, we compare to and follow the metric defined in Thorne et al. (2018). NoScoreEv is a simple classification accuracy that only considers the correctness of the verification label. On the other hand, ScoreEv is a stricter measure that also considers the correctness of the retrieved evidence. Hence, it is a more meaningful measure because it considers the classification to be correct only if appropriate evidence is provided to justify the classification.

4 Experimental Results

4.1 Oracle Performance

Document Retrieval: As shown in Table 2, our count-based re-rank strategy outperforms the TF-IDF method.

Take $k = 1$ as an example, we achieve 60.99% recall using only one document, which is ∼30% higher than TF-IDF approach. Given that the recall upper-bound of re-rank is 0.8886 (k=100), our method manages to retrieve near the limit by just retrieving a few documents. Note that there is

k	DR results			RTE results				
	Macro Recall	Macro Precision	F1	Accuracy		Evidence		
				ScoreEv	NoScoreEv	Precision	Recall	F1
2	0.729	**0.383**	**0.498**	**0.412**	**0.541**	**0.169**	0.655	**0.269**
3	0.768	0.270	0.396	0.407	0.534	0.166	0.672	0.267
4	0.791	0.209	0.328	0.407	0.535	0.164	**0.676**	0.264
5	**0.805**	0.170	0.279	0.397	0.529	0.161	0.675	0.260

Table 5: Effect of de-noising DR modules on RTE score

th	ES results			RTE results				
	Macro Recall	Macro Precision	F1	Accuracy		Evidence		
				ScoreEv	NoScoreEv	Precision	Recalll	F1
0.2	**0.653**	0.275	0.353	0.405	0.540	0.337	**0.629**	0.439
0.4	0.607	0.349	**0.406**	0.418	**0.542**	0.481	0.586	0.528
0.6	0.535	**0.368**	**0.406**	**0.424**	0.525	0.618	0.517	**0.563**
0.8	0.413	0.330	0.348	0.416	0.484	**0.772**	0.400	0.527

Table 6: Effect of de-noising ES modules on RTE score

a trade-off between the document recall and the noise ratio (i.e. low precision). As shown in Table 5, $k = 2$ with a low recall but high precision and F1 has the highest accuracy. This means DR that can effectively leverage this trade-off (therefore, high F1) performs the best. Therefore, we select $k = 2$ for our full-pipeline.

Evidence Selection: In Table 3, the trained neural ranker achieves a promising recall of 96.8%. In the case of $l = 5$, our neural ranker can perform 5% better than the TF-IDF method. Here, we use fixed $l = 5$ results for fair comparison with TF-IDF method. The ratios in Table 3 are the ratio of negative samples we tried to train the fake task. For example, 1:4 means that four negative sentences are sampled for each true evidence sentence. The results further give supports to our assumption that using unbalanced up-sampling actually help train our neural ranker. Along with performance gain, we also achieve a drastic gain in inference time by around 65 times from 3.57 seconds to 0.055 seconds for each sample.

The full-pipeline results for different values of th is shown in Table 6. Similar to document retrieval, having a high F1 score is the most important factor in assuring high full-pipeline performance. This is because providing a succinct set of evidence makes the verification task easier for the RTE module. Therefore, we choose DA_{rank}+NER model with $th = 0.6$ for the full-pipeline.

Recognizing Textual Entailment: The oracle classification accuracy for RTE is shown in Table 4. The MLP is a simple multi-layer perceptron using TF and TF-IDF cosine similarity between the claim and evidence as features as shown in Thorne et al. (2018). The highest accuracy achieved is 79.9% using DA_{rte} with NER information, thus, we further evaluate the full-pipeline accuracy on this setting.

4.2 Full pipeline Performance

Combining each of our improved pipeline modules using $k = 2$, $th = 0.6$, the full pipeline results are shown in Table 7. Our proposed framework can achieve 42.43% in ScoreEv and 52.54% in NoScoreEv, which outperforms DR_{tfidf}+DA by 11.55% and 2.10%, respectively. The evidence retrieval F1 in our full framework is 56.3%, which is improved promisingly by 38.80%.

5 Related Work

Prior work (Vlachos and Riedel, 2014; Ciampaglia et al., 2015) have proposed fact-checking through entailment from knowledge bases. Some works have investigated fact verification using PolitiFact data (Wang, 2017; Rashkin et al., 2017) or FakeNews challenge (Pomerleau and Rao). Most closely related to our work, Thorne et al. (2018) addresses large-scale fact extraction and verification task using a pipeline approach. In addition, question answering (Dang et al., 2007; Chen et al., 2017a; Ryu et al., 2014; Ahn et al., 2004) and task-oriented dialog systems (Dhingra et al., 2017;

Model	Label Accuracy (%)		Label			Evidence F1
	ScoreEv	NoScoreEv	Precision	Recall	F1	
DR_{tfidf} + MLP *	21.80	38.75	0.500	0.387	0.310	0.175
DR_{tfidf} + DA *	30.88	50.44	0.530	0.520	0.517	
Proposed	**42.43**	**52.54**	**0.533**	**0.527**	**0.523**	**0.563**

Table 7: Full-pipeline evaluation on the test set using $k = 2$ and $th = 0.6$. The first and the second one (with *) are the baselines from Thorne et al. (2018).

Madotto et al., 2018) also have similar aspects to these works, although aiming at a different goal.

Other fields that are related to the particular individual modules of our system are the following: Document and evidence retrieval for identifying text segments and documents to support a given claim (Salton and Buckley, 1987; Le and Mikolov, 2014; Cartright et al., 2011; Bellot et al., 2013; Rinott et al., 2015). Recognizing textual entailment that aims to determine whether a hypothesis *h* can justifiably be inferred from a premise (Dang et al., 2007; Bowman et al., 2015; Parikh et al., 2016; Chen et al., 2017b; Glockner et al., 2018). In some of these work (Rinott et al., 2015; Rashkin et al., 2017), the lexical and linguistic features are leveraged to further improve the performance.

6 Conclusion

In this paper, we extend the pipeline framework for fact-checking and propose a neural ranker for evidence selection. Our experiments show that the usage of lexical tagging is helpful in simplifying the task and improving the generalization ability. Moreover, reducing noise in the input of RTE module, by de-noising the DR and SR modules, appears to be crucial for improving the overall performance. As a result, our ranker outperforms the TF-IDF method by 38.8% in evidence retrieval F1, with 65 times faster inference speed, achieving a promising performance in a large-scale fact extraction and verification dataset.

References

David Ahn, Valentin Jijkoun, Gilad Mishne, Karin Müller, Maarten de Rijke, Stefan Schlobach, M Voorhees, and L Buckland. 2004. Using wikipedia at the trec qa track. In *TREC*. Citeseer.

Patrice Bellot, Antoine Doucet, Shlomo Geva, Sairam Gurajada, Jaap Kamps, Gabriella Kazai, Marijn Koolen, Arunav Mishra, Véronique Moriceau, Josiane Mothe, et al. 2013. Overview of inex 2013. In *International Conference of the Cross-Language Evaluation Forum for European Languages*, pages 269–281. Springer.

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

Marc-Allen Cartright, Henry A Feild, and James Allan. 2011. Evidence finding using a collection of books. In *Proceedings of the 4th ACM workshop on Online books, complementary social media and crowdsourcing*, pages 11–18. ACM.

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017a. Reading wikipedia to answer open-domain questions. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1870–1879, Vancouver, Canada. Association for Computational Linguistics.

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

Giovanni Luca Ciampaglia, Prashant Shiralkar, Luis M Rocha, Johan Bollen, Filippo Menczer, and Alessandro Flammini. 2015. tx. *PloS one*, 10(6):e0128193.

Hoa Trang Dang, Diane Kelly, and Jimmy J Lin. 2007. Overview of the trec 2007 question answering track. In *Trec*, volume 7, page 63.

Bhuwan Dhingra, Lihong Li, Xiujun Li, Jianfeng Gao, Yun-Nung Chen, Faisal Ahmed, and Li Deng. 2017. Towards end-to-end reinforcement learning of dialogue agents for information access. In *Proceddings of ACL*.

John Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(Jul):2121–2159.

Max Glockner, Vered Shwartz, and Yoav Goldberg. 2018. Breaking nli systems with sentences that require simple lexical inferences. *arXiv preprint arXiv:1805.02266*.

Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *International Conference on Machine Learning*, pages 1188–1196.

Andrea Madotto, Chien-Sheng Wu, and Pascale Fung. 2018. Mem2seq: Effectively incorporating knowledge bases into end-to-end task-oriented dialog systems. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1468–1478. Association for Computational Linguistics.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

Ankur P Parikh, Oscar Täckström, Dipanjan Das, and Jakob Uszkoreit. 2016. A decomposable attention model for natural language inference. *arXiv preprint arXiv: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.

Dean Pomerleau and Delip Rao. Fake news challenge.

Hannah Rashkin, Eunsol Choi, Jin Yea Jang, Svitlana Volkova, and Yejin Choi. 2017. Truth of varying shades: Analyzing language in fake news and political fact-checking. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2931–2937.

Ruty Rinott, Lena Dankin, Carlos Alzate Perez, Mitesh M Khapra, Ehud Aharoni, and Noam Slonim. 2015. Show me your evidence-an automatic method for context dependent evidence detection. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 440–450.

Pum-Mo Ryu, Myung-Gil Jang, and Hyun-Ki Kim. 2014. Open domain question answering using wikipedia-based knowledge model. *Information Processing & Management*, 50(5):683–692.

Gerard Salton and Chris Buckley. 1987. Term weighting approaches in automatic text retrieval. Technical report, Cornell University.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. FEVER: a large-scale dataset for fact extraction and verification. In *NAACL-HLT*.

Andreas Vlachos and Sebastian Riedel. 2014. Fact checking: Task definition and dataset construction. In *Proceedings of the ACL 2014 Workshop on Language Technologies and Computational Social Science*, pages 18–22.

William Yang Wang. 2017. "liar, liar pants on fire": A new benchmark dataset for fake news detection. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 422–426, Vancouver, Canada. Association for Computational Linguistics.