Fact Extraction and Automatic Claim Verification

Swapnil Kishore **Rohit Lalchand Vishwakarma** **Elizabeth Soper**

Abstract

We built an end-to-end system to automatically verify claims. The system extracts Wikipedia articles as evidence and classifies the claim based on BERT representations of the claim and evidence. Our best-performing system achieves a label accuracy of 45.5%, a 2 point improvement over our baseline system.

1 Introduction

In today's world, the circulation of news is instant and widespread. All it takes for information about an event at one end of the globe to reach the other end is a Twitter trend or a YouTube video. Due to this system, information can pass through several intermediary sources before reaching an individual and naturally gets altered in the process. Facts are diluted or corrupted and rumors originate in this way. It can be difficult to identify the truth among the noise of false claims. In this project, we developed an end-to-end system that verifies claims by extracting evidence related to them from Wikipedia pages. Based on the collected evidence, our system decides whether the claim is supported or refuted by the evidence, or whether there is not enough information to decide. The FEVER (Fact Extraction and Verification) dataset was created to train and evaluate systems on this task. In the following report we first outline the FEVER task, give an overview of the current state-of-the-art, describe a baseline system and our novel system, and finally analyse its performance.

2 FEVER Dataset and Shared Task

The FEVER (Fact Extraction and VERification) dataset consists of 185,445 claims manually classified as 'SUPPORTED', 'REFUTED', or 'NOTENOUGHINFO'. The purpose of this dataset is to evaluate an end-to-end system that performs the task of evidence extraction and claim verification. The dataset is divided into training data and testing data. The data contains the following fields:

1. Id: The id of the claim
2. label: One of {'SUPPORTED', 'REFUTED', 'NOT ENOUGH INFO'}
3. claim: The text of the claim
4. evidence: A list of evidence relevant to the claim.

Along with the FEVER dataset comes a predefined corpus of over 5 million pre-processed Wikipedia pages from a June 2017 dump, which should serve as the source of all evidence for the task.

FEVER task is conducted annually as a competition. It is composed of three subtasks, carried out in sequence: document retrieval, sentence selection, and recognizing textual entailment. Figure 1 below shows this process. In the document retrieval step, Wikipedia articles relevant to the claim are identified. In the next step, sentences are selected from the gathered documents that are most suitable to serve as evidence for (or against) the claim. Finally, based on the selected evidence sentences, the claim is labelled as 'SUPPORTED,' 'REFUTED,' or 'NOT ENOUGH INFO.'

The FEVER scoring system uses classification accuracy and evidence recall to evaluate systems. Points are awarded for accuracy only if the correct evidence is found. Only the first 5 predicted evidence sentences are selected for scoring. The rest are discarded without penalty. The scorer also produces other diagnostic scores such as F1 and precision. These are only used to rank two submissions with equal FEVER scores.

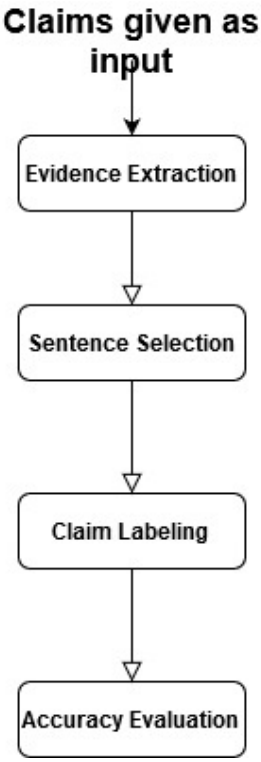

Figure 1: Flowchart of the FEVER task

3 Previous Work

The FEVER dataset and shared task were created by Thorne et al. (2018), who also outline a baseline system. Their baseline uses the DrQA system for evidence extraction. The system uses TF-IDF vector representations to choose relevant evidence. They retrieve documents relevant to each claim by finding the k documents with the highest cosine similarity to the claim. They then apply the same strategy at the sentence level; sentences from the relevant documents are ranked by cosine similarity to the claim. After selecting the relevant evidence, Thorne et al. (2018) compare two methods for classifying the claim: 1) an MLP with one hidden layer, which takes TF-IDF representations of claim and evidence as input; and 2) a decomposable attention model between the claim and evidence text. The decomposable attention model resulted in a significant improvement over a simple MLP. (52.09% vs 41.86% classification accuracy on the dev set). Their best performing system achieves 50.91% classification accuracy on the test set and an F1 score of 17.47% on identifying the correct evidence.

BERT (Bidirectional Encoder Representations from Transformers) is a versatile attention-based architecture that generates representations from unlabeled text (Devlin et al., 2018). BERT is trained on two objectives: 1) masked language modeling and 2) next sentence prediction. The models can be fine-tuned for a wide range of other NLP tasks, without significant task-specific modifications. While the training process is quite computationally expensive, several pretrained models are publicly available, and BERT or BERT-based architectures have become the SOTA in a wide variety of downstream NLP tasks (including question answering, text classification, natural language inference). Based on Thorne et al. (2018)'s positive results using decomposable attention, BERT seems like a logical next step in improving performance in claim verification.

Soleimani et al. (2020) apply BERT to the FEVER task, and achieve second place on the 2018 FEVER shared task. They use two fine-tuned BERT models: one for sentence selection and the other for claim classification (see Figure 2 below). For sentence selection, each potential sentence is input into BERT along with the claim and the resulting representation is classified as evidence or non-evidence. For claim classification, each piece of evidence is input with the claim to BERT and classified as supported, refuted, or not enough information. Their final system had a FEVER score of 69.66. Notably, by using BERT for

sentence selection, their system achieved a new state of the art in evidence recall, 87.1%.

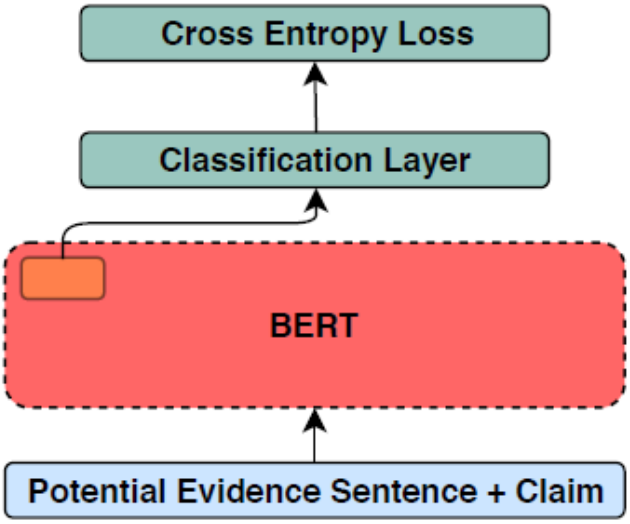

Figure 2: Sentence Selection/Claim Verification using BERT (Soleimani et al 2019)

Nie et al. (2019) created a homogenous Neural Semantic Matching Network Model for the FEVER task. In this homogeneous model, the same process is used for all the three subtasks. For document retrieval, they generalize its steps into keyword matching of the claim with the document title and its first sentence and for all other "disambiguative" documents they are using NSMN (Neural Semantic Matching Network). Also, they use page-view frequency to cut down the search space. Finally, the top k documents are selected for further processing. In the sentence selection phase, all the sentences are passed to NSMN to compare with the claim and the top five sentences are selected for the claim verification process. For claim verification, the NSMN is used with additional token-level features such as WordNet and 5-dimension real value embeddings to encode any unique number token, Normalized Semantic Relatedness Score. Their final system had a FEVER score of 66.14.

4 System Description

In the following section we outline our approach to the FEVER task. First, we describe our baseline system, then explain the steps we took to improve on the baseline.

4.1 Baseline System

Our baseline for document retrieval is using the 'wikipedia' library available in Python, which wraps around the MediaWiki API. We extract the noun phrases from each claim using the 'noun_phrases' component of TextBlob library, and pass those noun phrases as input to the command for retrieving pages. For our baseline, we retrieve the top three Wikipedia pages most relevant to the claim.

For sentence selection, we searched for the sources obtained in the Wikipedia corpus provided by the FEVER task. The Wikipedia corpus is a dump of pre-processed Wikipedia pages from June 2017. Here "pre-processed" means that a subset of sentences have automatically been selected from each of the pages. The chosen evidence must come from these sentences. We ranked the sentences in each source according to TF-IDF score. Then the top scoring sentence from each source is chosen. For this purpose a simple TF-IDF script in Python was used which calculates the scores for each sentence and returns the top scoring sentence.

For claim classification, we used the pretrained BERT-base model to create representations for each sentence. Each sentence is first tokenized by BERT's wordpiece tokenizer, then a [CLS] token is added before the first token and a [SEP] token is added after the last token. In this format, each claim and evidence sentence is passed through the BERT model, resulting in continuous-valued sentence representations. Finally, we concatenate the BERT representations for each claim and all its evidence sentences,

System	Label Accuracy	Precision	Recall	F1	FEVER score
Baseline	.435	.421	.082	.137	14.23
Similarity score: NPs	.414	.439	.112	.178	18.53
Similarity score: full text	**.455**	.467	.132	.206	21.20
Combined score: NPs	.428	.452	.121	.191	19.72
Combined score: full text	.446	**.485**	**.135**	**.211**	**22.06**

Figure 3: Summary of results on all tested systems

and use this as input to train a logistic regression classifier. For our baseline system, we trained the classifier on 10,000 examples.

4.2 Final System

We took several steps toward improving upon this baseline. For the document retrieval component, we tried two alterations. The first was passing the full text of the claim as input to the MediaWiki API, instead of just the noun phrases. The second strategy that we were able to implement for document retrieval was to use Python's built-in sequence matcher to find the most relevant Wikipedia pages for the claim. The title of each Wikipedia page was compared to the text of the claim and a score for each Wikipedia page is generated. Then the three highest scoring pages are selected. Since for this purpose the sequence matcher needs to generate a score for each of the over 5 million documents available, this method is extremely time-consuming and requires extensive resources. For this reason, thorough evaluation was not possible for this method.

In the sentence selection component of our system, the follow-up strategy we implemented was to calculate an overall score for each sentence by adding the raw TF-IDF score for each sentence to the cosine similarity score for that sentence and the claim. To generate the cosine similarity score the tfidf_vectorizer from sklearn and the sparse function from scipy are used. We test our results using the cosine similarity score alone to rank candidate sentences, as well as using the sum of the raw TF-IDF score (as in our baseline) and the cosine similarity score.

As the final step of the baseline system was the strongest, we made no major changes to our strategy for this component in the final system. The main improvement for this step was training on 15,000 examples instead of 10,000 examples. This was the most we could train on given our resource constraints.

5 Results and Discussion

In this section, we report and analyze the results of our system. Figure 3 summarizes the results of each of the systems we tested. 'Full text' vs 'NPs' refers to whether the system used the full text of a claim to retrieve documents, or only its noun phrases. 'Similarity score' represents systems where sentences are selected based on only the cosine similarity between the candidate sentence and the claim. 'Combined score' represents systems where sentences are selected based on the sum of the raw TF-IDF score and the similarity score. Although the highest overall label accuracy was achieved by using just the similarity score ('Similarity score: full text'), using the combined score and the full text yielded higher scores in precision, recall, F1, and the overall FEVER score. Using the full text of the claim and evidence sentences resulted in an increase in performance for both similarity score and combined score.

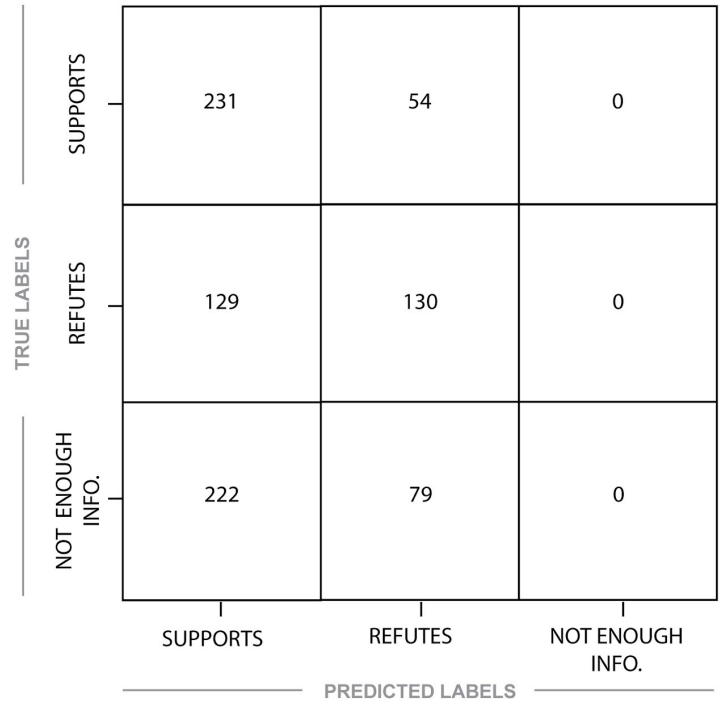

Figure 4: Confusion matrix, Final system

Figure 4 shows a breakdown of how our classifier labelled the test data. The classifier has 81.1% accuracy labelling true claims ('SUPPORTS'), 50.2% accuracy labelling false claims ('REFUTES') and 0% accuracy labelling unverifiable claims ('NOT ENOUGH INFO'). This confusion matrix makes obvious a key issue with our system: it never labels a claim 'NOT ENOUGH INFO' on the test data. This is because unverifiable claims in the training data had 'null' evidence, but our system chose the top 3 most relevant sentences for every claim, regardless of how relevant those sentences were to the claim, and so our classifier tried to categorize every claim that had any evidence as either 'SUPPORTS' or 'REFUTES.' We discuss a solution to this in the next section.

Also worth noting is the large discrepancy between precision and recall across all of our implemented systems. With our best-performing system ('Combined score: full text') we were able to extract at least one of the correct documents for just 17.8% of examples. But when the correct evidence was found, we were able to recognize textual entailment and label the claim with relatively high accuracy. Figures 5 and 6 show examples of our system's output for two claims: one was correctly classified, the other was incorrectly classified.

In the first example (Figure 5), our evidence is consistent with the ground truth evidence (though it includes fewer total sentences), and the label is correct. In the second example (Figure 6), our evidence is not at all relevant to the claim, and is consequently labelled incorrectly. Clearly, the relevance of the extracted evidence is crucial to the accuracy of the classifier. In fact, when tested using the ground truth evidence, our classifier label accuracy increases from 44.6% to 77.6%. Extracting the relevant evidence for a claim has proved to be the more challenging component of this task.

Ideally both the recall and the precision of a FEVER system would be high. In practice, however, precision should be a higher priority for fake news identification than recall. If our system says a claim is true, we should be very confident it is actually true. Since our goal is to prevent misinformation, having high precision and low recall is better than vice versa.

CLAIM: "Edgar Wright is a person."

Ground truth evidence: ["Edgar Howard Wright -LRB- born 18 April 1974 -RRB- is an English director , screenwriter , producer , and actor .", "He is best known for his comedic Three Flavours Cornetto film trilogy consisting of Shaun of the Dead -LRB- 2004 -RRB- , Hot Fuzz -LRB- 2007 -RRB- , and The World 's End -LRB- 2013 -RRB- , made with recurrent collaborators Simon Pegg , Nira Park and Nick Frost .\tShaun of the Dead\tShaun of the Dead\tHot Fuzz\tHot Fuzz\tSimon Pegg\tSimon Pegg\tNira Park\tNira Park\tNick Frost\tNick Frost", "He also collaborated with them as the director of the television series Spaced .\tSpaced\tSpaced", "He also co-wrote , produced and directed the 2010 film Scott Pilgrim vs. the World .\tScott Pilgrim vs. the World\tScott Pilgrim vs. the World", "Along with his friends Joe Cornish and Steven Moffat , he co-wrote Steven Spielberg 's The Adventures of Tintin : The Secret of the Unicorn .\tJoe Cornish\tJoe Cornish (filmmaker)\tSteven Moffat\tSteven Moffat\tSteven Spielberg\tSteven Spielberg", "Wright and Cornish co-wrote the screenplay for the Marvel Studios film Ant-Man , which Wright was intended to direct before departing the project .\tAnt-Man\tAnt-Man (film)"]

True label: “SUPPORTS”

Our evidence: ["Edgar Howard Wright -LRB- born 18 April 1974 -RRB- is an English director , screenwriter , producer , and actor .", "11"]

Our label: “SUPPORTS”

Figure 5: Correctly-labelled Claim

CLAIM: "Savages was exclusively a German film."

Ground truth evidence: ["Savages is a 2012 American crime thriller film directed by Oliver Stone .\tOliver Stone\tOliver Stone\tSavages\tSavages (2010 novel)\tcrime\tCrime film\tthriller\tThriller (genre)", "Savages is a 2012 American crime thriller film directed by Oliver Stone .\tOliver Stone\tOliver Stone\tSavages\tSavages (2010 novel)\tcrime\tCrime film\tthriller\tThriller (genre)"]}

True label: “REFUTES”

Our evidence: ["23\tChristian churches were also oppressed , with many leaders imprisoned .", "; unless they are credited as co-production partners -RRB- nor any direct-to-video releases , TV films , theatrical re-releases , or films originally released by other non-Disney studios .", "Africa\tAfrica\n2\tThe film was shot over a period of three years by Gualtiero Jacopetti and Franco Prosperi , two Italian filmmakers who had gained fame -LRB- along with co-director Paolo Cavara -RRB- as the directors of Mondo Cane in 1962 ."]

Our label: “SUPPORTS”

Figure 6: Incorrectly-labelled Claim

6 Directions for Future Work

Based on the results reported above, we propose several ideas for future work on the FEVER task. On the document retrieval component, it would be worth trying to search the FEVER corpus directly instead of using MediaWiki API, which runs a search of the full, current version of Wikipedia. Since the FEVER corpus is more limited than the full Wikipedia, and because the ground truth evidence all comes from this corpus, finding a way to efficiently search the FEVER corpus directly would likely improve our recall.

On the sentence selection component, using Named Entity Recognition instead of extracting noun phrases from the claim might yield better results. Another crucial addition to this component is defining a threshold for relevance. If no sentence is above threshold, the claim should be automatically labelled 'NOT ENOUGH INFO.' This would solve our classification problem, where every claim gets labelled either 'SUPPORTS' or 'REFUTES.' One final idea is to use BERT representations, like we did in the claim classification component of our system, to select sentences. This would be similar to the strategy of Soleimani et al. (2020).

Finally, on the claim classification component of our system, two improvements could be made. First, training on a larger portion of the training dataset might boost performance (resource constraints limited us to training on just 15k of 145k available examples). Second, fine-tuning the pretrained BERT model might improve our performance, allowing for our classifier to learn subtler task-specific patterns, since BERT was originally trained on a masked language model objective. Fine-tuning could be especially helpful for handling cases where the decision relies on information from two or more separate sentences.

7 Conclusion

In this report we have described our end-to-end system for evidence extraction and claim verification. Our best-performing system achieved an accuracy of 45.46% and a peak evidence F1 score of 21.7. As misinformation become easier and easier to manufacture and propagate, the problem of identifying fake news will only become more critical. Thus continued work on the FEVER task and systems like the ones described here is important now and going forward.

Group Member Contributions

Swapnil Kishore implemented the Document Retrieval and Sentence Selection components of our system. Elizabeth Soper implemented the Claim Classification component of our system and wrote up the reports. Rohit Lalchand Vishwakarma created our web app and worked on improving the performance of the Document Retrieval component.

References

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. (Mlm), 2018. URL `http://arxiv.org/abs/1810.04805`.

Yixin Nie, Haonan Chen, and Mohit Bansal. Combining Fact Extraction and Verification with Neural Semantic Matching Networks. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33: 6859–6866, 2019. ISSN 2159-5399. doi: 10.1609/aaai.v33i01.33016859.

Amir Soleimani, Christof Monz, and Marcel Worring. BERT for Evidence Retrieval and Claim Verification. pages 359–366, 2020.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. FEVER: A large-scale dataset for fact extraction and verification. *NAACL HLT 2018 - 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference*, 1:809–819, 2018.