IR Term Project: Evidence Retrieval for Fact Verification

Abstract

The increasing concern with misinformation has stimulated research efforts on automatic fact checking. The recently released FEVER dataset introduced a benchmark fact verification task in which a system is asked to verify a claim using evidential sentences from Wikipedia documents. In this report, we propose a model to solve the fact verification task.We have used a document retrieval model that employs an entity extraction model that is based on constituency parsing with tf-idf scoring based evidence retrieval.We further employ BERT, an effective pre-trained language representation to get better accuracy.For the fine tuning purpose we have used the RoBERTa model available in the HuggingFace model hub as "roberta291 large-mlni".We achieved an accuracy of 68% in document retrieval and 98.88% in evidence retrieval.

1 Introduction

The ever-increasing amounts of textual information available combined with the ease in sharing it through the web has increased the demand for verification, also referred to as fact checking.Here we focus on verification of textual claims against textual sources. Fact or claim verification is a two-step process. First, we retrieve supporting or refuting evidence related to a claim. Then based on the set of evidence snippets, the task is to determine whether the claim is true or false.

This fact verification finds demand in various domains of the world.We show its relevance through various applications in the real world.With the rise of fake news and misinformation, fact verification is more important than ever when reading news articles. Checking the sources and verifying the accuracy of the information presented can help avoid spreading false information. Many people use social media to share news and information, but not all of it is accurate. Before sharing a post or article, it's important to fact-check the information to ensure that it's true.When it comes to medical information, it's essential to verify the accuracy of the information before making any decisions about your health. This might include checking the credentials of the source, verifying statistics or data, and consulting with medical professionals. In the world of academic research, fact verification is essential to ensuring the validity and reliability of findings. Researchers must carefully check their sources and data to ensure that their work is accurate and trustworthy. In legal cases, fact verification can be critical to determining guilt or innocence. Lawyers and judges must carefully examine evidence and testimony to ensure that it is accurate and reliable.

These are just a few examples of situations where fact verification can be important. In general, any time we encounter information that we are unsure about, it's a good idea to fact-check it .

2 Related Works

The baseline model consists of a simple pipelined system comprising three components: document retrieval, sentence-level evidence selection and textual entailment. Each component was evaluated in isolation through oracle evaluations on the development set.The document retrieval component returned the k nearest documents for a query using cosine similarity between binned unigram and bigram Term FrequencyInverse Document Frequency (TF-IDF) vectors.The simple sentence selection method ranked sentences by TF-IDF similarity to the claim and sorted the most-similar sentences first and tuned a cut-off using validation accuracy on the development set. For recognizing textual entailment a multi-layer perceptron (MLP) with a single hidden layer which ses term frequencies and TF-IDF cosine similarity between the claim and evidence as features was used.

The GEAR framework employed a three-step pipeline with components for document retrieval,

sentence selection and claim verification to solve the task.For the document Retrieval potential entities were extracted from the claim and the relevant Wikipedia documents were found.For sentence Selection, sentences were selected according to their decreasing order of relevant scores calculated using the Hinge Loss Function. They were filtered with a threshold to alleviate the noises. The claim verification with Gear consisted of three parts.A Sentence Encoder where BERT model was used to represent in evidence reasoning graph. Evidence nodes contain information from claim to guide the flow of messages among them. A Evidence Reasoning Network which is a fully-connected evidence graph of T layers. Features are linear combination of normalized attention coefficients calculated using MLP. An Evidence Aggregator which finds the final state using information from evidence graph. The aggregator may utilize different aggregating strategies and three aggregators were suggested in the framework. These are the Attention Aggregator Max Aggregator and the Mean Aggregator.

In the UKP Athene , in the document retrieval task entity linking was used . The constituency parser from AllenNLP was used after which every noun phrase was considered as a potential entity mention.The MediaWikiAPI3 is used to search through the titles of all Wikipedia articles for matches with the potential entity mentions found in the claim. The MediaWiki API uses the Wikipedia search engine to find matching articles. The top match is the article whose title has the largest overlap with the query.Sentence selection is achieved by extending the Enhanced Sequential Inference Model(ESIM) to generate a ranking score on the basis of two input statements.The modified ESIM takes as input a claim and a sentence. To generate the ranking score, the last hidden state of the ESIM is fed into a hidden layer which is connected to a single neuron for the prediction of the ranking score. In order to classify the claim as Supported, Refuted or NotEnoughInfo, the five sentences retrieved by sentence selection model described previously was used. For the classification, another extension to the ESIM was proposed, which can predict the entailment relation between multiple input sentences and the claim.

The Neural Semantic matching networks developed neural models for the task.The encoding layer encoded the input sentences using LSTM.The alignment layer aligned the two sentences to same dimensions.By using the aligned and the original encoded representations were combined.The matching layer performed semantic matching on compound representation using the BILSTM.A Max-pooling was performed row wise on the matching sequences. The two sequences p, q and abs(p-q), p.q(element wise) were fed to the output layer for final predictions. For document and sentence retrieval, predictions were two scores whether we should consider document/sentence or not.Similarly for the fact verification, three scores were the outputs(for SUPPORT,REFUTE,NOT ENOUGH INFO).It Integrated the evidence retrieval and claim verification by passing evidence retrieval score vectors to CV Model. This helped the model focus better on some pieces of evidence more based on semantic matching strength.It also concatenates WORDNET features into token-level features(on top of Glove/ELMO). Relations like antonyms,hyponyms, hypernyms, etc were used to compare the claim and the set of evidences.

.

3 Dataset

The dataset used is the Fever 2018 shared-task dataset which is a large-scale dataset containing 185,445 claims, each of which comes with several evidence sets. An evidence set consists of facts, i.e. sentences from Wikipedia articles that jointly support or contradict the claim. On the basis of (any one of) its evidence sets, each claim is labeled as Supported, Refuted, or NotEnoughInfo if no decision about the veracity of the claim can be made. Supported by the structure of the dataset, the FEVER shared task encompasses three sub-tasks that need to be solved. From the Fig. 1, we can observe a example from the dataset.

"SUPPORTED" Example	
Claim	The Rodney King riots took place in the most populous county in the USA.
Evidence	(1) The 1992 Los Angeles riots, ***also known as the Rodney King riots*** were a series of riots, lootings, arsons, and civil disturbances that ***occurred in Los Angeles County***, California in April and May 1992. (2) ***Los Angeles County***, officially the County of Los Angeles, ***is the most populous county in the USA***.
"REFUTED" Example	
Claim	Giada at Home was only available on DVD.
Evidence	(1) ***Giada at Home*** is a television show and first ***aired*** on October 18, 2008, ***on the Food Network***. (2) ***Food Network*** is an American ***basic cable and satellite television channel***.

Figure 1: Example from the Fever Dataset

4 Techniques/Methods

4.1 Document Retrieval

Given a claim, the main task is to find the most relevant Wikipedia articles i.e. documents with respect to the claim. The key steps involved in it are:

4.1.1 Mention Extraction

There can be many approaches to finding the most relevant parts in a claim sentence. One of the methods is to use a named-entity recognition model that focusses on specific types of entities like person, organization, location etc. Hence, we tried using a NER model for the keyword extraction using the libraries spacy and stanza. However, as seen from the results, these models can't give all the necessary entities as they are having very specific categories. Hence, we employ an entity extraction model that is based on constituency parsing using the Stanza library. After parsing the claim using Stanza we consider every noun phrase as a potential entity that can be used to search for relevant documents. In order to do so, we extract the constituency tree of each claim and find the position of the words which are having labels **"NP" or "NML"**. Using these noun-phrases, we can extract the candidate articles in the next step.

4.1.2 Document Extraction

We created a database called **Fever.db** using Wikipages.jsonl files from the Fever-shared task 2018. This database contains the titles of all the Wikipedia pages with their respective contents, which was used for retrieving the sentences. We used the MediaWiki API to search each of the noun phrases to extract the top Wikipedia pages. The MediaWiki API is based on a matching algorithm that tests for maximum overlap with the noun-phrase. Using these predicted Wikipages, we can search our **Fever.db** database to retrieve the contents of each of them. In this way, for each of the claims we retrieve the relevant documents together with their lines.

4.2 Evidence Retrieval

The next step is to find the most relevant sentences from all the retrieved documents. For this we need to use some form of relevance scoring mechanism for each of the sentences in these documents. We have used two approaches for this task namely, a TF-IDF based scoring mechanism that uses the Sklearn library. All the documents in the database are used to construct the tf-idf weighting values, which can be used later to find the similarity between the claim and each sentence. The measure of similarity in each (claim,sentence) pair is calculated by using cosine similarity between the vectors.

4.2.1 Disadvantages of Tf-Idf

The tf-idf model does not use the context in each of the sentences. It is only based on count-based measures on the set of documents. This is not desirable as tf-idf cannot correctly represent the semantic meaning of the sentences. In addition, it is not able to react to changes in the ordering of the sentences. Thus, in order to extract better semantic representations to turn to more powerful models.

4.2.2 Bert-based models

Pre-trained language models such as BERT, GPT, ELMO, etc have proven to perform exceedingly well on many NLP tasks. BERT(Devlin et al.,2019) employs a bidirectional transformer and well designed pre-training tasks to fuse bidirectional context information. Since these large language models are pretrained on a large corpus of data consisting of Wikipedia(2.5 billion words) and Google's BooksCorpus(800 M words). These large informational datasets can to BERT's deep knowledge not only in English but also in multilingual settings as well. BERT uses two objectives during pretraining namely MLM(Masked Langauge Models) and NSP(Next Sentence Prediction). We have used the 'bert-base-uncased' model for the purposes of the evidence retrieval task.

4.2.3 Evidence Retrieval Using BERT

We use the Pytorch library to perform the evidence retrieval using BERT. The BERT for sequence classification model was used with two labels corresponding to relevant(1) and non-relevant(0). The pretrained model is finetuned on our specific fever data. For the purposes of finetuning the model we need samples corresponding to (claim,sentence) pairs and the outputs(0/1) pairs. We extract these samples from train.jsonl dataset. The positive samples(relevant ones) can be directly taken from the ground truth evidences of the fever data. However, for the negative samples we need to extract them using some scheme. However it is not prefarable to directly sample any line from the fever.db database as this can result in very vague examples that are

not useful for finetuning. Hence, we get the top documents from the MediaWiki API and select any document from the list of results. Similarly, the sentence is randomly sampled from the above. This ensures that the negative samples that are extracted are relevant and are not completely unrelated to the claim. Next we use the retrieved (claim,sentence) pairs through a pretrained tokenizer to get the input tensors which is a concatenation of the claim and the candidate sentence.The **max length** of the sentences are taken to be 256. Sentences shorter than this are padded with a special <PAD> token and those longer are truncated. The next step taken is to use mini-batch gradient descent to fine-tune the model. For this purpose, we initialize Dataloaders. We use cross-entropy as loss function. The optimizer used is the Adam optimizer with a learning rate of 2e-5. The pretraining was run for 5 epochs. The fine-tuned model is then used on the retrieved documents and each candidate sentence and the probability of sentence being relevant is extracted. This is then used to find the top-k evidences for the given claim. The retrieved evidences with the claims can then be fed into the Fact Verification model to output one of the 3 labels (SUPPORTS,REFUTES,NOT ENOUGH INFO).

4.3 Fact Verification

Evidence Retrieval module provides us with a list of evidences corresponding to a claim. With the help of this claim-evidences pair, we need to classify the claim into three classes - SUPPORTS, REFUTES and NOT ENOUGH INFO. The fact verification module basically performs this classification.

4.3.1 Fine-tuning

For this task, we choose the RoBERTa model available in the HuggingFace model hub as "roberta-large-mlni". It is one of those language models, which is very powerful with the textual entailment tasks.

We can load the model using python code and then provide it with a claim-evidences pair to get back the predicted label. The challenge here is that the model is specifically fine-tuned to the Multi-Genre Natural Language Inference (MNLI) task, and might not perform optimally on the FEVER task. Therefore, we need to fine tune the model on the FEVER task.

In order to fine tune the model, we need three things - the claim, the evidence and the labels. Therefore, we first generate a fine-tuning dataset in a CSV file from the "train.jsonl" file and "fever.db" database. In "train.jsonl", we have a claim, the name of the Wikipedia pages and the exact id of the sentence forming the evidence. Using this page name and id, we extract the exact sentences from the fever database. Then we save them in the CSV file.

In case of NOT ENOUGH INFO, the evidences are absent. So, we follow the Nearest Page approach to generate evidences in this case. In this approach, we first find the noun phrases present in the sentence. Then, we use "wikipedia" library in python to extract the closest wikipedia page to that noun phrase. From each such page, we extract the first sentence. All these extracted sentences act as evidences for NEI case.

With the help of the generated fine-tuning dataset, we fine-tune and update the parameters using AdamW optimizer. This gives us the updated parameters of the RoBERTa model.

4.3.2 Testing the classifier

We can now directly load the fine-tuned model and use it to make textual-entailment predictions. The model takes claim-evidences pair as input and returns the probabilities of the classes - SUPPORTS, REFUTES and NEI. The class with maximum probability is assigned to the sample. Based on the predicted and ground truth labels, the accuracy of the classifier is observed.

5 Experiments

5.1 Document Retrieval and Evidence Retrieval

1. We computed the accuracy of the retrieval methods. It is shown below:

Retrievals	Accuracy
Document Retrieval	68%
Evidence Retrieval using TF-IDF	60%

2. For a particular claim example Fig.2, the corresponding retrieved evidences from the TF-IDF model are shown in Fig.3 and retrieved evidences from the BERT model are shown in Fig.4.

3. From the plot 5, we can observe the cross-entropy loss diminishing with every epoch

while fine-tuning the BERT model for evidence retrieval.

```
In [362]: claim_test_lines[0]
Out[362]: 'Nikolaj Coster-Waldau worked with the Fox Broadcasting Company.'
```

Figure 2: Claim example

```
In [361]: retrieved_evidences_tfidf[0]
Out[361]: [['Fox_Broadcasting_Company', 0],
           ['Fox_Broadcasting_Company', 13],
           ['Fox_Broadcasting_Company', 1],
           ['Nikolaj_Coster-Waldau', 7],
           ['Fox_Broadcasting_Company', 10]]
```

Figure 3: Retrieved evidence set (TF-IDF)

```
In [365]: retrieved_evidences_bert[0]
Out[365]: [['Fox_Broadcasting_Company', 0],
           ['Nikolaj_Coster-Waldau', 0],
           ['Nikolaj_Coster-Waldau', 3],
           ['Fox_Broadcasting_Company', 10],
           ['Nikolaj_Coster-Waldau', 2]]
```

Figure 4: Retrieved evidence set (BERT)

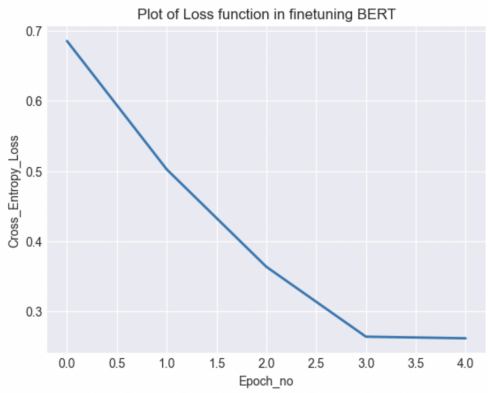

Figure 5: BERT Fine-tuning loss

5.2 Fact Verification

First of all, we see the output on a randomly chosen claim from the development dataset.

```
Claim:  One of the leads in Transformers: Age of Extinction is an American rapper.
Evidences:
1 :  It stars Mark Wahlberg , with Peter Cullen reprising his role as the voice of
Optimus Prime , as the lead roles .
2 :  Mark Robert Michael Wahlberg -LRB- born June 5 , 1971 -RRB- is an American actor
, producer , businessman , former model , and rapper .
Label:  SUPPORTS
```

Figure 6: Claim, evidence and predicted label

Now, we observe and compare the accuracies of the fact verification task using various models - RoBERTa Large MNLI, Fine-tuned RoBERTa Large MNLI, and Facebook BART MNLI in the following table :

RoBERTa	RoBERTa Fine-tuned	Facebook BART mnli
64%	83%	65%

Here, we find that the accuracy of vanilla Facebook BART mnli model is higher compared to vanilla RoBERTa model. Also, Facebook BART mnli model supports 1024 max tokens as input as compared to 512 tokens in RoBERTa. Therefore, we believe that if we fine-tune the Facebook BART mnli model and then use it, it will give even better accuracy than 83%. We are yet to fine-tune this BART model and measure its accuracy for fact verification task.

The accuracies of individual classes for the fine-tuned RoBERTa Model are as follows:

SUPPORTS	REFUTES	NEI
92%	86%	96%

Thus, we can see that using the NearestPage Approach results in a very high accuracy score for NEI Case. Another reason for high accuracy of NEI case is that the sentence retrieval part is not involved in the NEI case as we select the first sentence from the wikipedia page itself. So, the final error is the combined error of only the document retrieval and the fact verification modules, hence higher accuracy compared to other classes.

6 Analysis and Future Work

6.1 Analysis

1. There may be spelling mistakes in the claim provided by the user. So, Spelling correction is a method where we can provide alternatives to which the user was referring. This corrected query would be a better way in order to find the best possible results.

2. There may be some grammatical errors in the claim which would not be efficient queries because there are various examples where the noun phrases extracted would be inappropriate.

3. We have all the hyperlinks given at the end of each sentence in the Fever.db database. While doing evidence retrieval, we can also include lines of the hyperlinked pages by using some similarity measure between claim and the beginning of the hyperlinked page. This can improve the performance of the model in Multi-hop settings.

4. The accuracy of document retrieval is 68%. The main reason behind it is the disambigua-

tion in the titles of the Wikipedia pages. For example, our document retrieval model extracts the page "Homeland" but the ground truth page is "Homeland (TV series)". This will be calculated as wrong document retrieval.

6.2 Future Work

1. Instead of using the preprocessed wikipedia pages dataset, we can use the wikipedia library of python to directly access the content of any wikipedia page. It would make the system more robust and user-friendly.

2. The page view frequency can be taken under account for the analysis of the relevance score of the resulting Wikipages.

3. We can use WordNet and ConceptNet features for important tokens in the retrieved evidence and also the claim and include these in the evidence retrieval and the fact verification step.

4. In the fact verification part, we can use some large language-models which can support large size inputs of tokens 1024 or more. This would improve the results when the input sequence is of large size. But it would require more computational and storage resources. For example, we can use "gpt-x2".

7 References

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

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.

Jie Zhou, Xu Han, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. 2019. GEAR: Graph-based Evidence Aggregating and Reasoning for Fact Verification.

Yixin Nie, Haonan Chen, Mohit Bansal, Combining Fact Extraction and Verification with Neural Semantic Matching Networks

David Wadden, Kyle Lo, Lucy Lu Wang, Arman Cohan, Iz Beltagy, and Hannaneh Hajishirzi. 2022. MultiVerS: Improving scientific claim verification with weak supervision and full-document context

8 Work Distribution

Name	Experiments
Shreyash Vaish	Fact Verification Module, Nearest Page Approach, Comparison of RTE Models
Saraf Parth Vikrant	Document Retrieval, Evidence-Retrieval using TF-IDF
Debrup Das	Evidence Retrieval using BERT Model, Future Work
Kamalesh Garnayak	Analysis of related works, Code review