# Automatic Fact Verification

Sahaj Dhingra 960448 , Swapnil Shailee 952247

## Abstract

The Automatic Fact Verification task is used for verifying the claims by labelling them as 'Supports', 'Refutes' or 'Not Enough Info' by extracting facts from the given dataset. In this paper, we present our fact verification approach which extracts the facts using the TF-IDF model and uses the cosine similarity to return the best matching page identifier. For sentence retrieval we introduce the entity matching technique which preprocesses the claims by rephrasing them. Named Entity Recognition(NER) tags and Ngram similarity is then used to match the sentences of the paragraph and the preprocessed claims, thus classifying them into labels.

## 1 Introduction

Information Extraction is a significant research domain and the outputs of such systems enable many natural language technologies such as text summarization and question answering system. It helps in retrieving relevant data from a large dataset including social media websites and online dictionaries. It has various applications including but not limited to business intelligence, resume harvesting , media harvesting, patent search and fact verification. Due to massive increase in the amount of flawed and deceptive content on the Internet, there has been high demand for information evaluation through fact-checking to verify false claims that are present on the Internet. Automatic fact verification system is based on extracting the relevant information from the dataset to verify a claim. The system is built using the Wikipedia data-set which is the text corpus comprising of 109 text files which corresponds to approximately 3.5 GB of information. Training and development data as mentioned in table 1, have been used for evaluating and improving the performance of the system. The system is then run to predict the labels of the test set.

| Datasets | Number of claims |
|---|---|
| Training | 145449 |
| Development | 5001 |
| Test | 14997 |

Table 1: Claims in datasets

## 2 Background

The system is built using the concepts of Information retrieval and Natural language processing which consist of 3 main steps:

- Page Retrieval
- Sentence Selection
- Recognising Textual Entailment

Various approaches have been previously used for this purpose. (Vlachos and Riedel, 2014) approach took advantage of the various websites available for fact verification such as PolitiFact for claim verification on a very small dataset. In their approach, to verify the claim, various non machine-readable justifications were needed which made it a challenging task for the Machine Learning methods.

Another approach which overcame the above challenge was used by (Wang, 2017) where the labels of the claims were generated not by the considering the justification but by taking the person's name and the metadata of the claim. Though the approach was good and efficient, but the labels generated could not verified against any credible sources. In this paper we used the entity linking approach used by (Hanselowski et al., 2018) which addresses both the problems mentioned above. In this approach, in the pre-processing step we extract the entities from the best matching page from the Wikipedia data and compare it with the named entities of the given claims. Though the pre-processing step is time consuming but once the entity extraction is done it runs efficiently on the large data set of the claims. Moreover, the claims can be justified by looking at the predicted evidence containing the page and sentence identifier. Taking the approach of Recognizing Textual Entailment used by (Thorne et al., 2018), we also introduced an enhancement to our approach by rephrasing the claims by using antonyms and negating them before matching them with the extracted entity.

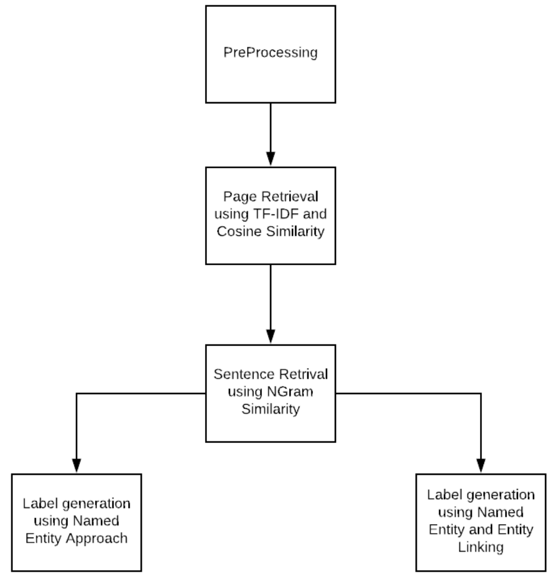


Figure 1: Architecture of Automatic Fact Verification.

# 3 Methodology

The main modules involved for building the system are the following:

1. Preprocessing
    - Computing unigram and bigram of wikipedia Dataset: The system retrieves all the unigrams and bigrams from the dataset by using tokenizing, lowercasing and stemming techniques of the NLTK library.
    - Computation of TF-IDF: The TF-IDF model is created and stored for further processing.
    - Entity Extraction: The entity dictionary is created by extracting the entities by pos-tagging the data and considering the proper nouns as entity names from the given wikipedia data set which is store for further processing.
    - Rephrasing the claims: Claims are POS tagged and verbs are negated and antonyms of the adjectives are used to rephrase the claim and store it in processed claim dictionary.
2. Page Retrieval
    - Computation of the top K ranking Documents :The top K documents for the claim vector is generated based on the TF-IDF model. We enhanced the model used by (Loper and Bird, 2002) where instead of just using unigram TF-IDF we use a combination of unigram and bigram TF-IDF in our approach. The cosine similarity

is then used to find the best matching page identifier out of the top 5 pages.

3. Sentence Selection

   - N-gram similarity: The bigram similarity between the claim vector and sentences from the best matching page is calculated to find the most relevant sentence.

4. Recognising Textual Entailment: Two main approaches are followed here namely Named Entity Recognition Tagging Approach and Entity Linking Approach.

   - Approach 1 using Named Entity Recognition Tagging: Using Spacy NER, NER tags for best matching sentence and claim are compared for label generation. In case of a match, SUPPORT is the label result otherwise, REFUTE is returned as the label result.
   - Approach 2 using Named Entity Recognition Tagging and Entity Linking Approach: The claims from the processed claims dictionary are matched with the text of the entities. If the match is there with the normal claim then label is "Support" for the claim otherwise "Refutes" is returned as result.

# 4 Analysis and Results

For our analysis we compare both the approaches with each other. From the results in table 2 we clearly see that when using the Named Entity Recognition only the label accuracy is just below the baseline of 40%. This is mainly because the chosen similarity threshold is insignificant in determining the label correctly. This is easily overcome by enhancing the method and adding the entity linking approach. This time the results are independent of the threshold as the labels are calculated by comparing the claims from the processed claims dictionary and the best matching sentences. An example can be seen where the claim "Magic Johnson did not play for the Lakers." is compared with the sentence "He played point guard for the Lakers for 13 seasons" gave wrong label because of the threshold specified.Another example can be seen where the claim "Saxony is the tenth smallest German state." when matched with the sentence "Saxony is the tenth largest of Germany 's sixteen states , with an area of 18,413 km2 , and the sixth most populous , with 4 million people." predicts wrong label in the first approach. Moreover, the correctness of approach 2 can also be justified by increase in the sentence precision from 20.52 to 30.88. Due to the limited resources, we ran our system on a truncated test data to submit the results and got a relatively lower rank.

| Performance Metric | Value |
|---|---|
| Label Accuracy | 38.56% |
| Sentence Precision | 20.52% |
| Sentence Recall | 18.31% |
| Sentence F1 | 19.40% |
| Document Precision | 30.17% |
| Document Recall | 30.17% |
| Document F1 | 30.17% |

Table 2: Results of Approach 1

| Performance Metric | Value |
|---|---|
| Label Accuracy | 44.30% |
| Sentence Precision | 30.88% |
| Sentence Recall | 40.31% |
| Sentence F1 | 35.59% |
| Document Precision | 30.17% |
| Document Recall | 30.17% |
| Document F1 | 30.17% |

Table 3: Results of Approach 2

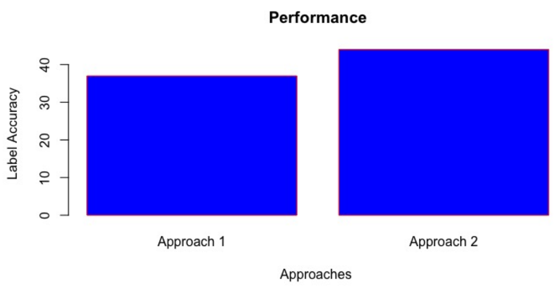


Figure 2: Label Accuracy Performance of Approaches


```
{"claim": "Men in Black II stars an actor born on a boat.", "label":
"REFUTES", "evidence": [["Men_in_Black_3", 2], ["Men_in_Black_3", 8],
["Men_in_Black_3", 1], ["Men_in_Black_3", 0]]}
```


Figure 3: Numerical Error

# 5 Error Analysis

The error analysis of the system can be described as follows:

- Numerical Errors:The system is inefficient in matching Roman numerals with numbers in claims and evidences by the Named Entity tagging of the system.In the given Claim in figure 3, the system matches II with Men in Black 3 evidence instead of Men in Black 2.Better NER Taggers would solve numerical errors in the system.

- Document Matching Errors: The system returns incorrect evidences for certain claims due to high similarity in certain documents which resulted in decreased document F1 score.For the given claim in figure 4, the system finds high similarity for terms Aristotle and Athens in the following evidences which is incorrect. The system could not link Aristotle and Athens. Improvised and efficient information retrieval and similarity can be incorporated to solve these errors. Improvised entity linking approach can be used to enhance the performance of the system.


```
{"claim": "Aristotle spent time in Athens.", "label": "REFUTES", "evidence":
[["Elias_Mariolopoulos", 7], ["Elias_Mariolopoulos", 8],
["Elias_Mariolopoulos", 5], ["Elias_Mariolopoulos", 9],
["Elias_Mariolopoulos", 26]]}
```


Figure 4: Document Error


```
{"claim": "The Cretaceous ended.", "label": "SUPPORTS", "evidence":
[["Cretaceous", 8], ["Cretaceous", 2], ["Cretaceous", 5], ["Cretaceous",
0]]}
```


Figure 5: Evidence Error

- Evidence Errors:The system correctly supports the claim and provide correct evidence but one single evidence is sufficient to support the claim but the system returns multiple evidences. This decreases the sentence F1 score of the system. It can be observed in the given claim in figure 5.Better similarity techniques like Allen NLP or Stanford CoreNLP can be incorporated to solve these kind of errors.

# 6 Conclusion

In this paper we tried to solve the ongoing Fact Extraction and verification problem by using entity extraction approach along with the named entity recognition. In order to keep contributing in this area, we would like to keep on enhancing our approach. This can be done mainly by using more rephrasing techniques for the claims like finding the synonyms, substituting similar and dissimilar words and rephrasing the sentence keeping the meaning same. Moreover, use of the better trained Named Entity Tagger could be used for better results. We strongly believe that though this task is challenging but it's still feasible.

# 7 References

KDE Group,. 2014. Fact checking: Task definition and dataset construction | BibSonomy.

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.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. FEVER: a large-scale dataset for Fact Extraction and VERification.

William Yang Wang. 2017. "Liar, Liar Pants on Fire": A New Benchmark Dataset for Fake News Detection.

Michael Mausam, Robert Schmitz, Stephen Bart, and Oren Soderland. 2012. Open Language Learning for Information Extraction. Technical report, July.

Silviu Cucerzan. 2017. Large-Scale Named Entity Disambiguation Based on Wikipedia Data.