# Document Retrieval and Fact Verification

Agnibha Sinha, Daksh Varshney, Dhruv Rathi, Kaushal Jadhav
Prof. Somak Aditya, Neeraj Saini, Ankan Mullick

***Abstract*— Document retrieval and Fact Verification**
**In this project, we make use of the FEVER dataset to retrieve documents according to the query for the purpose of fact verification. The FEVER Dataset has 145449 claims. The claims are classified as SUPPORTED, REFUTED or NOTENOUGHINFO. After removing the claims with labels as NOTENOUGHINFO we get 109810 claims. From the list of documents, we tokenize the query and the claims and apply cosine similarity to rank them. We then use LSTM model to verify the fact from the retrieved documents. The accuracy of our LSTM model is 0.79.**

## I. Problem Statement

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. Our primary focus in this project is to retrieve documents based on the query.

## II. Motivation

The amount of online data has grown at least as quickly as the speed of the computer [1]. With the emergence of various online news sources and the social media the spread of misinformation is becoming a major problem leading to real-world violent events that threaten public safety and other harmful societal, political and economical impacts. Either intentionally or unintentionally, with the use of internet individuals and organizations can reach large audiences to circulate the claims which may be false. Such fake news often manipulates the public opinion, spreads unnecessary fear and creates conflicts. Example of tempting power and profit associated with fake news is during the U.S. presidential election dozens of teenagers in the Macedonian town of Veles created fake news on social media and became wealthy. The ever-increasing amount of information available combined with the ease in sharing through the web has led to the demand of verification referred to as fact verification [7]. Although fact verification has received a lot of attention in the field of journalism, it is also important for other domains and applications like verification of medical claim, product reviews, news and scientific publications. Manual fact verification is a time-consuming and intellectually demanding process that is performed by trained professionals. Recently, many websites have emerged to allow expert-based fact-verification, for example, PolitiFact, FactCheck, HoaxSlayer etc [11]. PolitiFact takes three editors to judge whether a piece of news is real or not. Therefore manual verification has limited use due to its inability to handle the huge volume of misinformation on the internet and to react quickly. Consequently, there has been a lot of demand to provide automation to reduce the human time and burden in performing fact verification [8]. Automated fact verification is a challenging and complex task which requires finding the textual evidence followed by evidence reasoning and entailment [9]. Reasoning provides a scientific principle to describe why the evidence supports the claim. Entailment in natural language processing is a directional relation between text fragments. This automated fact verification has attracted attention of researchers. Different computational approaches have been presented to automate verification task with a common goal of identifying false or misleading information. For the retrieval of relevant evidence from a collection of documents, existing systems generally utilize traditional sparse retrieval which may have poor recall, especially when the relevant passages have few overlapping [10]. Many tasks only consider the interaction between an article body and headlines and they do not consider the need to find appropriate evidence as a human would do [11]. There is always a need to work for finding an effective automated solution to combat the online falsehoods when the amount of information required to be checked is growing exponentially [7].

## III. Related Work

Literature shows the increased demand for fact verification has stimulated rapid progress in developing various computational methods for evidence retrieval and fact verification. In most studies, fake news detection is formulated as a classification or regression problem [12]. 1) Classification: The most common way is to formulate the fake news detection as a binary classification problem. However, categorizing all the news into two classes (fake or real) is difficult because there are cases where the news is partially real and partially fake. Mainly, a category for the news which is neither completely real nor completely fake is set as additional classes. When using these datasets, the expected outputs are multi-class labels and those labels are learned as independent labels. (Rashkin et al., 2017; Wang, 2017). However, as obtaining reliable labels requires a lot of time and labor, semi/weakly-supervised and unsupervised methods are used (Rubin and Vashchilko, 2012; Bhattacharjee et al., 2017). 2) Regression: Fake news detection can also be formulated as a regression task where the output is a numeric score of truthfulness. This approach is used by Nakashole and Mitchell (2014). Usually, evaluation is done by calculating the difference between the predicted scores and the ground truth scores or using Pearson/Spearman correlations. However, since the available datasets have discrete ground truth scores, converting the discrete labels to numeric scores is a challenge. These days such systems use machine learning to compare claims against evidence and see whether they are true or not. Machine Learning Models The majority of existing research uses supervised methods while semi-supervised or unsupervised methods are less commonly used.

## IV. Technique and Experiments

Beyond the goal of simply comparing a claim against given, relevant evidence is the broader task. Hence there is a need for some source from which the potential evidence can be extracted. Once a source is identified, a system may analyze the text from a claim and use that to extract the necessary evidence to either support or refute that claim [7].

**Document Retrieval:**

The complete code containing the proposed approach of using Cosine Similarity and Word to Vector for document retrieval and LSTM Model for fact verification is written in Google Colaboratory IPYNB Notebooks. At first we pre-processed the data-set. We used the json inbuilt package to input the data from jsonl file available at (https://fever.ai/dataset/fever.html). We created separate text files for storing claims and corresponding labels, IDs and evidence Wikipedia URLs. We removed the claims which do not have evidence URLs listed. We found that out of the total 145449 claims, 35639 claims had to be removed. Thus, the training set is 0.755 of the dataset.To implement Word to Vector model[23] stemming and tokenization was found to be necessary. Hence, we tokenized the claims using nltk package. We removed stop-words and punctuation while tokenizing claim sentences. We also extracted URL links into a separate list for future use. After pre-processing we stored the dataset in a seperate CSV file. For evidence extraction we first used Word2Vec model of gensim library and then used cosine similarity.We used the Continuous Bag of Words (CBOW) model. We use cosine similarity for mapping a given test query to a train query. Using the mapping relevant evidences are found. We use cosine similarity as a metric to rank the documents. We also implemented the fever scorer to get the scores of the documents retrieved by our model.

**Examples:**

Query: nikolaj
Files retrieved: ['Nikolaj_Coster-Waldau', 'Nicole_Kidman', 'Nicki_Minaj', 'International_relations', 'Jennifer_Aniston']

Query: roman
Files retrieved: ['Room_93', 'Ron_Perlman', 'Marilyn_Monroe', 'Abraham_Lincoln', 'Paramore_-LRB-album-RRB-']

Query: history
Files retrieved: ['Yardie_-LRB-novel-RRB-', 'Hypothyroidism', 'Cirrhosis', 'Ridley_Scott', 'Sarcoidosis']

A few random queries were tried on the trained model. The predicted documents could be retrieved successfully along with the cosine similarity scores. A few examples have been demonstrated in the following video.
https://youtu.be/i7TBx1Fp8q0

**LSTM:**

After the documents were retrieved, we applied LSTM model to classify the claims into 2 classes, support and refute. The model was trained using the claims present in claimsss.csv file. It had a total of 1,09,810 claims with labels. Out of this, the number of claims supported were 80,035. We divided the dataset into test and train datasets in a 20-80 percent split. The number of trainable parameters in the LSTM were 387,601 which were trained in 3 epochs. More epochs were not performed due to the limitations on time required to run the code. Finally, we got the predictions and the values of precision, recall and f1 score.

| Labels | Precision | Recall | f1-score |
|---|---|---|---|
| refutes | 0.79 | 0.27 | 0.4 |
| supports | 0.79 | 0.97 | 0.87 |

A few random claims were tried on the trained model. The predicted claims could be printed successfully along with the probabilities. A few examples have been demonstrated in the following video.
https://youtu.be/sxjeRJ7NX1c

**Examples:**

Liberty bonds were bought in the United States - Support(correct)

Roman Atwood is a content creator - Refute(correct)

Sophie Turner was born in the 1990's - Support(correct)

Stranger Things is set in Bloomington, Indiana - Refute(incorrect)

Roman Atwood is a content creator - Support(correct)

Massachusetts is close to Rhode Island - Support(correct)

Massachusetts is close to Rhode Island - Refute(incorrect)

## V. Analysis and Future Thoughts

Fact verification has become an ongoing struggle for as long as people have been spreading information. Cosine-similarity do not perform well on original data, which is highly dimensional, noisy, and sparse. The performance of LSTM on documents having label refute is not great as there are very less training examples of the same. We applied the LSTM model for classifying the claims into support or refute. Instead of it, BERT model could have been used to get better results.

## WORK DISTRIBUTION

| Team Member | Contribution |
|---|---|
| Agnibha Sinha | Code for fact verification using LSTM<br>Made the report and helped in presentation |
| Daksh Varshney | Made presentation and helped in report |
| Dhruv Rathi | Code for Data pre-processing and tokenization<br>Made presentation and helped in report |
| Kaushal Jadhav | Code for document retrieval using cosine similarity<br>Code for FEVER scorer for ranking<br>Made the report and helped in presentation |

## REFERENCES


[1] Christopher Manning, Prabhakar Raghavan, Hinrich Schütze, "Introduction to Information Retrieval", Cambridge University Press

[2] Giannis Bekoulis, Christina Papagiannopoulou and Nikos Deligiannis, (In press) "A Review on Fact Extraction and Verification", ACM Computing Surveys, Volume 55, Issue 1, January 2023, pp. 1–35

[3] Shyam Subramanian and Kyumin Lee, "Hierarchical Evidence Set Modeling for Automated Fact Extraction and Verification", Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), November 2020, pp. 7798-7809

[4] Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Tuo Zhao, "SMART: Robust and Efficient Fine-Tuning for Pre-trained Natural Language Models through Principled Regularized Optimization", Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Online, Association for Computational Linguistics, July 2020, pp. 2177–2190

[5] Jie Zhou, Xu Han, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li and Maosong Sun, "GEAR: Graph-based Evidence Aggregating and Reasoning for Fact Verification" Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, , August 2019, pp. 892–901

[6] Bernhard Kratzwald and Stefan Feuerriegel, "Adaptive Document Retrieval for Deep Question Answering", Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, November 2018, pp. 576–581

[7] James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal, "FEVER: a Large-scale Dataset for Fact Extraction and verification", Proceedings of NAACL-HLT, 2018, pp. 809–819

[8] L. Graves, "Understanding the Promise and Limits of Automated Fact-Checking", Technical Report, Reuters Institute, University of Oxford, February 2018.

[9] Fan Yang, Eduard Dragut, Arjun Mukherjee, "Improving Evidence Retrieval with Claim-Evidence Entailment", Proceedings of Recent Advances in Natural Language Processing, September 2021, pp.1553–1558

[10] Chris Samarinas, Wynne Hsu, and Mong Li Lee, "Improving Evidence Retrieval for Automated Explainable Fact-Checking", Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Demonstrations, Online, Association for Computational Linguistics, June 2021, pp. 84–91.

[11] Alessandro Bondielli and Francesco Marcelloni, "A Survey on Fake News and Rumour Detection Techniques", Information Sciences, Vol. 497, 2019, pp. 38–55

[12] Ray Oshikawa, Jing Qian, and William Yang Wang, "A Survey on Natural Language Processing for Fake News Detection", Proceedings of the 12th Language Resources and Evaluation Conference, March 2020

[13] Rashkin, H., Choi, E., Jang, J. Y., Volkova, S., and Choi, Y., "Truth of Varying Shades: Analyzing Language in Fake News and Political Fact-checking", Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, Sept. 2017, pp. 2931–2937

[14] Wang, W. Y., "Liar, liar pants on fire": A New Benchmark Dataset for Fake News Detection, Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, Vol. 2, July 2017, pp. 422-426

[15] Rubin, V. L. and Vashchilko, T., "Identification of Truth and Deception in Text: Application of Vector Space Model to Rhetorical Structure Theory", Proceedings of the Workshop on Computational Approaches to Deception Detection, 2012, pp. 97–106

[16] Nakashole, N. and Mitchell, T. M., "Languageaware Truth Assessment of Fact Candidates", Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, Vol. 1, June 2014, pp.1009–1019

[17] Shu, K., Mahudeswaran, D., Wang, S., Lee, D. and Liu, H., "Fakenewsnet: A Data Repository with News Content, Social Context and Dynamic Information for Studying Fake News on Social Media", arXiv preprint arXiv:1809.01286, 2018

[18] Bhattacharjee, S. D., Talukder, A. and Balantrapu, B. V, "Active Learning Based News Veracity Detection with Feature Weighting and Deep-shallow Fusion", IEEE International Conference on Big Data, 2017, pp. 556–565

[19] Hassan, N., Arslan, F., Li, C., and Tremayne, M., "Toward Automated Fact-checking: Detecting Checkworthy Factual Claims by Claimbuster", Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2017, pp.1803–1812

[20] Karimi, H., Roy, P., Saba-Sadiya, S., and Tang, J., "Multi-source Multi-class Fake News Detection", Proceedings of the 27th International Conference on Computational Linguistics, 2018, pp. 1546–1557

[21] Della Vedova, M. L., Tacchini, E., Moret, S., Ballarin, G., DiPierro, M., and de Alfaro, L., "Automatic Online Fake News Detection Combining Content and Social Signals", 22nd Conference of Open Innovations Association (FRUCT), 2018, pp. 272–279

[22] Dagan, I., Dolan, B., Magnini, B. and Roth, D., "Recognizing Textual Entailment: Rational, Evaluation and Approaches–erratum", Natural Language Engineering, Vol. 16, Issue 1, Jan 2010, pp. 105-121

[23] Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean, "Efficient Estimation of Word Representations in Vector Space", Proceedings of International Conference on Learning Representations, 2013, arXiv:1301.3781v3 [cs.CL]