Questcheck: Validating Unverified Texts from Social Media using Machine Reading Comprehension

Yash Muchhala
Department of Information Technology
Sardar Patel Institute of Technology, Mumbai, India

Sai Nimkar
Department of Information Technology
Sardar Patel Institute of Technology, Mumbai, India

Atul Kamble
Department of Information Technology
Sardar Patel Institute of Technology, Mumbai, India

Dr. Radha Shankarmani
Department of Information Technology
Sardar Patel Institute of Technology, Mumbai, India

Abstract

Social Media has become a tool for mass communication. However, some pieces of information that float around in the different mediums are not backed by facts and lack credibility. The spread of such unverified information should be curbed as it has wide-spread results in spreading misinformation that has the potential to directly affect the opinions of people consuming such information. A lot of interest in research for classifying such unverified news has been seen in the last few years. Most of the aforementioned research directly dictates that the data sources be centralized, this not only is computationally resource intensive but also requires constant effort to make sure all verified sources of truth are up to date. Therefore, in order to efficiently classify unverified news, we present a stateless technique QuestCheck, which works on the principle of Machine Reading Comprehension by aggregating the questions generated from the unverified information and pooling for answers from verified news sources on the same topic. Our validation algorithm uses a custom metric to check whether the answers obtained from both sources, i.e. verified and unverified, have a semantic similarity. Ideal scenarios for this would be validating factual information pieces, statements by massively followed personalities, and official government policies amongst other things. We present a comprehensive explanation about QuestCheck, demonstrate the working with concrete real-world examples backed by data analysis along with a detailed evaluation metric using Natural Language Processing to discuss the effect of our technique on curbing the spread of misinformation.

Keywords: Stateless, Natural Language Processing, Machine Reading Comprehension, Semantic Similarity

I. INTRODUCTION

Misinformation on Social Media platforms consists of the deliberate insertion of malicious information, hoaxes or propaganda designed to influence the opinions of everyone that consumes information using the medium. Such misinformation is specifically manipulated to target audiences in order to influence political movements, form targeted opinions, and even affect voting patterns, amongst other things, all of which have the potential to support propagandas and ill-informed movements.

The spread of such unverified information should be curbed as it as wide-spread results in spreading misinformation that has the potential to directly affect the opinions of people consuming such information. A lot of interest in research for classifying such unverified news has been seen in the last few years. Most of the aforementioned research directly dictates that the data sources be centralized, this not only is computationally resource intensive but also requires constant effort to make sure all verified sources of truth are up to date. Therefore, in order to efficiently classify unverified news, we present a stateless technique QuestCheck, which works on the principle of Machine Reading Comprehension by aggregating the questions generated from the unverified information and pooling for answers from verified news sources on the same topic. Our validation algorithm uses a custom metric to check whether the answers obtained from both sources, i.e. verified and unverified, have a semantic similarity.

The background work, methodology, along with the results of the study of validating such unverified information are presented and discussed in this paper.

II. BACKGROUND AND RELATED WORK

A. Effects of Unverified Information on Public Perception

Unverified news has been demonstrated as an effective way of advertising a propagandistic agenda. It has had an influence on people's opinions as shown in [1]. It has negatively affected companies, businesses, organizations [2] and even individuals [3] resulting in defamation, physical injuries, and even death.

B. Machine Learning Algorithms for Fake News Detection

In an attempt to classify fake news using influence mining, T. Traylor et al [4] introduced a new technique namely Influence Mining to detect fake news with the help of Natural Language Processing and SciPy Toolkits. It studied and identified the technical linguistic patterns of the fake news and a classifier model was developed with supporting Machine Learning Grammar. This classifier used the attribution of the quote in the documents as the only factor for labeling the news. The Attribution Score or the A-Score Algorithm was devised to assign a final attribution score for all the documents containing quotes which are found from the results of the Machine Learning Classifier used before. The tool produced an accuracy of 69.4% in identifying real and fake news documents from the test set.

Another technique proposed by Kuriakose et al [5] was to identify the authenticity of the news using Artificial Neural Networks coupled with Sentiment Analysis. Their solution used a feed-forward artificial neural network architecture using a Rectified Linear Unit as an activate function to calculate a click-bait score for a news article in consideration. Further, thorough sentiment analysis is carried out using a Naïve Bayes Classification algorithm on the comments to validate or invalidate the news article.

C. Comparing Various Approaches To Classify Misinformation

Another study, by S. Gilda [6] evaluates the performance of different machine learning algorithms on a dataset acquired by Signal Media. It evaluates the performance of TF-IDF (Term Frequency-Inverse Document Frequency) and PCFG (Probabilistic Context-Free Grammar) both individually and combined on these below Machine Learning algorithms considering the same dataset.

1) Model Bounded Decision Trees,
2) Gradient Boosting,
3) Random Forest,
4) Stochastic Gradient Descent,
5) Support Vector Machines.

TF-IDF combined with the Stochastic Gradient Descent model predicts unverified articles as true or false with an accuracy score of 77.2%.

Lastly, in a study by Aphiwongsophon et al [7], the researchers demonstrate that the Naïve Bayes algorithm has the potential to classify fake news with an accuracy of 96.08%. The other two techniques were Neural Network-based and using Support Vector Machines accomplished an accuracy of 99.90%.

While highly accurate results have been achieved over already available datasets, it still remains a challenge to replicate these results on real-world data which could potentially be used as a reliable means of verifying such misinformation without human intervention.

III. METHODOLOGY

The methodologies used to research the classification of misinformation, the strategies behind every method and their implementation are discussed in this section.

QuestCheck, which works on the principle of Machine Reading Comprehension by aggregating the questions generated from the unverified information and pooling for answers from verified news sources on the same topic. Our validation algorithm uses a custom metric to check whether the answers obtained from both sources, i.e. verified and unverified, have a semantic similarity. Ideal scenarios for this would be validating factual information pieces, statements by massively followed personalities, and official government policies amongst other things.

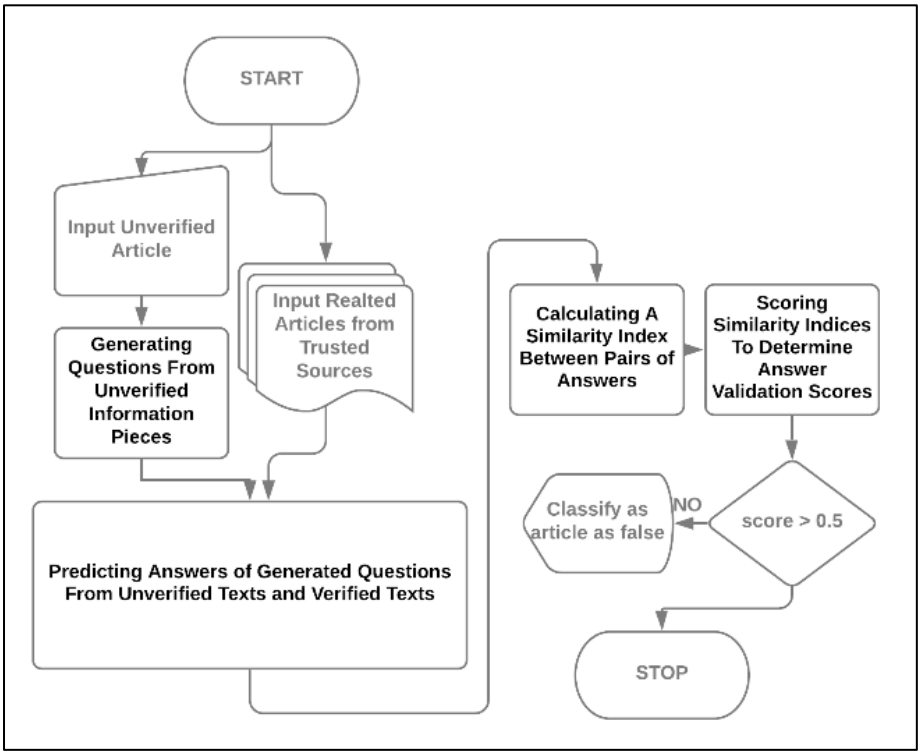

Fig. 1: Flowchart for the methodology of QuestCheck.

A. Generating Questions From Unverified Information Pieces

Our theory was based on the Sequence2Sequence generation of question-answer pairs from the SQuAD2.0 dataset. Generating exhaustive question-answer pairs from unverified texts would extract all information from the text as answers to corresponding questions. These answers further needed to be verified with answers predictively obtained from trusted sources with an input of the previously generated questions. We used the pre-trained Bidirectional Encoder Representation model (BERT) [8] fine-tuned on SQuAD2.0 specifically for question generation.

Another parameter for generating questions was to determine the number of questions to be generated. We determine the number of questions to be generated (N_q) as

$$N_q = \text{number of words (of unverified text)} \times K$$

Where K is defined as content density. K=0.5 works well for smaller texts, whereas any value between K=0.1 to K=0.5 for small to medium texts would be suitable as it would theoretically exhaust as sensible question-answer pairs that would be needed to check. A high K value would result in a higher N_q value, which while giving better accuracy would be more computationally expensive. So there's a tradeoff when determining an optimal K value. Also, it's worth noting that a higher K could probably result in an averaged accuracy as compared to an optimal value of K. Methods to improve obtaining an optimal N_q have been discussed in the later sections.

B. Predicting Answers of Generated Questions From Unverified Texts and Verified Texts

The questions generated from the unverified article are fed as input into a custom question answering machine learning model along with text scraped from trusted sources that are related to the news which is to be verified. In essence, question answering is just a prediction task — on receiving a question as an input, the goal is to identify the right answer from the given corpus. Machine Reading Comprehension tasks such as this have seen tremendous success when fine-tuned of a pre-trained language model [9]. In our proposed solution, we have trained the pre-trained BERT [7] model on SQuAD2.0 to generate answers for the posed questions on the input information.

The BERT training process uses the next sentence prediction. A pre-trained model based on Transformer [10] (BERT) uses next sentence prediction which is relevant for tasks like question answering. During training, the model receives pairs of sentences as input and learns to classify if the succeeding sentence is the sentence after the current one in the original text as well. So, given a set of questions and a corpus, the model predicts a begin and an end token from the given corpus that most probably fits as the answer to the question. Using this approach, our question-answering machine learning model can be trained by learning an additional of two tokens considered as vectors that denote the start and the end of the predicted answer.

C. Developing a Similarity Index between Pairs of Answers

The validation algorithm uses text semantic similarity to match answers obtained from the previous model discussed in 3.2. i.e. answers for each question from the unverified source compared to each answer from trusted sources for the same question under consideration. There are many ways text similarity can be done. Our model uses knowledge-based measures that quantify semantic relatedness of words using a semantic network.

WordNet [10], is one of the lexical databases designed for Natural Language Processing that groups English words into sets called synsets. Here we consider the problem of embedding entities and relationships of multi-relational data in low-dimensional vector spaces, its objective is to propose a canonical model which is easy to train, contains a reduced number of parameters and can scale up to very large corpora. Text Similarity between two pieces of text is computed by determining how similar both are in terms of context and surface closeness.

We used a Natural Language Processing pipeline in our text similarity scoring technique to determine text similarity with higher accuracy. When a semantic similarity is to be determined between two answers, first all stop words are removed from the text. Stopwords are words that are used as filler words and are of very less semantic importance, they're defined in the NLTK library. Second, all words are reduced to their root words using WordNet so that words which only differ in their tense, or plurality amongst other things are not treated differently. It is also called stemming. Finally, we get rid of all punctuations and case dis-similarities.

After the necessary pre-processing steps as discussed above, we use WordNet's Lemmatizer. Lemmatization in linguistics is the process of grouping together the inflected forms of a word so they can be analyzed as a single item, identified by the word's lemma, or dictionary form. The lemmatized words are then mapped with NLTK words using synsets. Synset is an interface that is packaged in NLTK to look up for words in WordNet. Synsets are the groups of words that have the same meaning or express similar concepts. Some of the words have only one Synset and some have several. The WuP Similarity is the measure of relatedness which considers the depths of the two synsets & depth of the LCS (Least Common Subsumer) in the WordNet. LCS is the most specific concept which is an ancestor of both synsets.

$$\text{WuP Similarity} = \frac{2 \times \text{depth(LCS)}}{\text{depth(synset1)} + \text{depth(synset2)}}$$

The WuP similarity score is always between 0-1 and the score is 1 when the two concepts are the same.

Finally, an average of all maximum similarities of each word-word pair is taken, and a value between 0 and 1 inclusive is found. Similarity index 1 denotes that both the texts are totally identical, and a similarity index of 0 denotes that two texts under consideration have no apparent semantic similarity shared between them.

D. Scoring Similarity Indices To Determine Answer Validation Scores

Now, each similarity index computed for an answer-answer pair needs to be evaluated alongside the other answer-answer pairs for each question. Not only that, but all computed values also have to be evaluated among all the question-answer-answer pairs. We use a simple weighted average where greater weights are given to texts with a lower similarity index. Our proposed use of simple weighted averages is due to the phenomenon that most of the answer-answer text pairs obtained from 3.2. would have a high similarity score, and only those having a misinformed answer would pertain to a lower similarity score.

Finally, taking the minimum of all the weighted averages from all answer-answer pairs obtained from each question, we evaluated the final scoring metric. However, this may fail when there are trusted sources related to the unverified text. Further improvements over our scoring metric are discussed in later sections.

IV. RESULTS AND DISCUSSIONS

Since this is only an experimental paper, we have tested our technique on an example from PolitiFact.com. PolitiFact.com is a website by the Poynter Institute for Media Studies, which is the owner of the International Fact-Checking Network. We discuss an example and the results of QuestCheck in this section

A. Example

Unverified: "Says the CDC now says that the coronavirus can survive on surfaces for up to 17 days."

Trusted Source: "2019 coronavirus can live for up to 3 hours in the air, up to 4 hours on copper, up to 24 hours on cardboard up to 3 days on plastic and stainless steel."

As per 3.1, generated questions from Unverified News with N_q=1. Generated question: "How long does it take the coronavirus to survive on surfaces?"

As per 3.2, answers to this question for the above-unverified news and trusted source are "17 days" and "3 days" respectively.

As per 3.3, we calculate the similarity index between the two answers. The similarity score calculated is 0.44 which is an indication that the answers do not match, and thus, the unverified news can be deemed as false.

V. FUTURE WORK

Our target with this research was to construct a novel method towards validating unverified information pieces. Further research could be focussed on improving the optimization of N_q such that it covers all the factually important details from the text while not jeopardizing the accuracy by averaging it out over a large number of values. Scoring metrics can also be optimized towards calculating a value more indicative of the conclusion. Such a metric can also be multi-dimensional to incorporate complex relationships between the question-answer-answer pairs' semantic similarity scores.

We will address these issues in a future research effort along with incorporating a robust dataset to exhaustively test and improve upon the current methods.

VI. CONCLUSIONS

1) This paper presented the results of a study to classify unverified news articles as true or false using a novel technique which is based on the principle of Machine Reading Comprehension.
2) It proposes a way to detect fake news by calculating a semantic similarity score for the answers of generated questions on the unverified news versus the pooled answers from verified sources of the same topic.
3) The technique used for the study is novel in this topic domain and is promising for further research.

REFERENCES

[1] M. Balmas, "When Fake News Becomes Real: Combined Exposure to Multiple News Sources and Political Attitudes of Inefficacy, Alienation, and Cyncism," Communic Res., vol. 41, no. 3, pp. 430-454, 2014.
[2] C. Silverman and J. Singer-Vine, "Most Americans Who See Fake News Believe It, New Survey Says," BuzzFeed News, 06-Dec-2016.
[3] C. Kang, "Fake News Onslaught Targers Pizzeria as Nest of Child-Trafficking," New York Times, 21-Nov-2016.
[4] Traylor, T., Straub, J., Gurmeet, & Snell, N. (2019). Classifying Fake News Articles Using Natural Language Processing to Identify In-Article Attribution as a Supervised Learning Estimator. 2019 IEEE 13th International Conference on Semantic Computing (ICSC).
[5] Kuriakose, A., Sebastian, D., Mathew, E. M., Mathew, H., & Er.Gokulnath, G. (2019). ALIKAH- A Clickbait and Fake News Detection System using Natural Language Processing. 2019 3rd International Conference on Trends in Electronics and Informatics (ICOEI).
[6] Gilda, S. (2017, December). Evaluating machine learning algorithms for fake news detection. In 2017 IEEE 15th Student Conference on Research and Development (SCOReD) (pp. 110-115). IEEE.
[7] Aphiwongsophon, S., & Chongstitvatana, P. (2018, July). Detecting Fake News with Machine Learning Method. In 2018 15th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTICON) (pp. 528-531). IEEE.
[8] Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. Bert: Pretraining of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805, 2018.
[9] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding with unsupervised learning. Technical report, OpenAI.
[10] George Miller. WordNet: An on-line lexical database. International Journal of Lexicography, 3(4), 1990. (Special Issue).