ClaimVerif: A Real-time Claim Verification System Using the Web and Fact Databases

Shi Zhi Yicheng Sun Jiayi Liu Chao Zhang Jiawei Han
University of Illinois at Urbana-Champaign, Urbana, IL, USA
{shizhi2,ysun73,jiayi4,czhang82,hanj}@illinois.edu

ABSTRACT

Our society is increasingly digitalized. Every day, a tremendous amount of information is being created, shared, and digested through all kinds of cyber channels. Although people can easily acquire information from various sources (social media, news articles, *etc.*), the truthfulness of most received information remains unverified. In many real-life scenarios, false information has become the *de facto* cause that leads to detrimental decision makings, and techniques that can automatically filter false information are highly demanded. However, verifying whether a piece of information is trustworthy is difficult because: (1) selecting candidate snippets for fact checking is nontrivial; and (2) detecting supporting evidences, i.e. stances, suffers from the difficulty of measuring the similarity between claims and related evidences.

We build ClaimVerif, a claim verification system that not only provides credibility assessment for any user-given query claim, but also rationales the assessment results with supporting evidences. ClaimVerif can automatically select the stances from millions of documents and employs two-step training to justify the opinions of the stances. Furthermore, combined with the credibility of stances sources, ClaimVerif degrades the score of stances from untrustworthy sources and alleviates the negative effects from rumor spreaders. Our empirical evaluations show that ClaimVerif achieves both high accuracy and efficiency in different claim verification tasks. It can be highly useful in practical applications by providing multi-dimension analysis for the suspicious statements, including the stances, opinions, source credibility and estimated judgements.

1 INTRODUCTION

With the increasingly digitalized process of our society, a tremendous amount of information is being created, shared, and digested by people on a daily basis. Information is accessible from everywhere, but in many scenarios, hardly would people know if the news they know is in accordance with the existing facts. According to [1], in the 2016 U.S. President Election, the average percentage of fake news received by American citizens is 0.92 for Trump and 0.23 for Clinton, while nearly half of the population believe these fake news are trustworthy.

In this paper, we demonstrate a real-time claim verification system that can verify the claim automatically using the documents from the Web and fact databases. Here, fact databases refer to recently emerging fact-checking websites, which collect unverified claims and provide review articles written by expert editors. Examples include Snopes.com, factcheck.org, and politifact.com. We extract the labeled *claims* and their *review articles* from fact-checking websites to formulate a set, named *fact database*.

To build such a system, the first key challenge is the difficulty in finding related snippets of a claim, due to the diversity of natural language expressions. A straightforward way is to learn a fact-checking classifier with the fact database. However, although the labels of the claims are provided, only a small proportion of sentences directly represent the opinions of a related article. Treating the whole article as the training documents would introduce irrelevant information. Thus, it is necessary but challenging to first obtain a set of relevant snippets — what we call *stances* — that explicitly represent the opinions to a claim.

The second challenge is that the user-given claim does not naturally co-exist with its review articles. Thus, it is essential to get the related documents from the Web. However, there is a gap between the human-crafted review articles of the fact database — which is specifically written for claim verification, and the report articles retrieved from the Web. Furthermore, the credibility of the information providers also plays an important role when using the stances. Even when an article expresses strong opinions to a claim, it needs to be neglected when the information provider frequently spreads fake news and rumors.

Rumor detection has been well studied for social media [2, 5, 10] and Internet data [3, 7, 8]. Meanwhile, there have also been tools [1] that verify the credibility of news by analyzing the trust factors, such as image analysis and Google's safe browsing API. However, these existing methods are not designed for analyzing the contents of the claim itself, and cannot address the above challenges.

To tackle these challenges, we develop an end-to-end claim verification system, called *ClaimVerif*, which can judge the truthfulness of a live query claim in real-time, and rationale its judgements with supporting evidences. The contributions of our system are summarized as follows: (1) We use the embedding-based method to extract the related snippets in order to capture the semantic similarity between the claim and candidate snippets. (2) We develop a two-step classifier to classify the opinion of the related articles with limited number of high-quality editorial review articles and sufficient web documents. (3) Our system implements a reweighing module to incorporate source quality of the information providers. (4) Our systems is optimized to support real-time claim queries which balances both accuracy and efficiency.

[1] https://devpost.com/software/fib

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

CIKM'17 , November 6–10, 2017, Singapore, Singapore
© 2017 Association for Computing Machinery.
ACM ISBN 978-1-4503-4918-5/17/11...$15.00
https://doi.org/10.1145/3132847.3133182

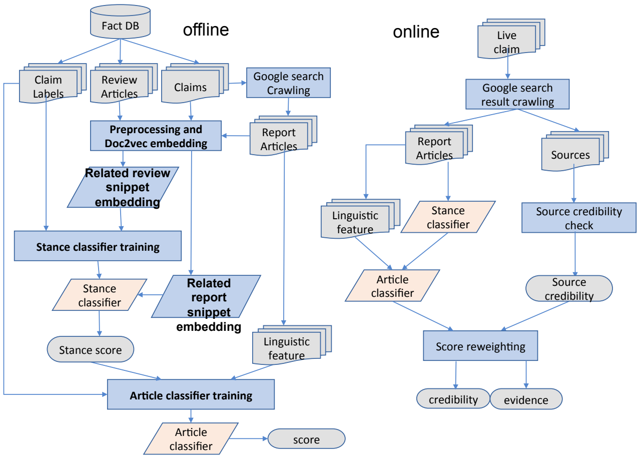


Figure 1: System Overview

2 SYSTEM IMPLEMENTATION

Fig. 1 shows the framework of ClaimVerif. As shown, it consists of an offline training pipeline and an online querying workflow. In the offline pipeline, different claims, review articles and report articles are fed into a stance classification module and a two-step training module. In the online part, we first regard the user-given claim as a query to retrieve related report articles from the search engine, and then we pass both the query claim and the articles into the trained modules as well as a credibility assessment module. We describe different components in the following.

2.1 Stance Extraction

The stance extraction component is designed to find the most related snippets to the claim. Extracting stances for claims is essential to our claim verification system, because a claim will be mainly judged by its associated stances in the online assessment stage. Generally, we extract stances for claims by first retrieving relevant report articles from commercial search engines, and then using the report articles to generate stances for claims in the fact database.

HTML preprocessing. We first feed claims into commercial search engine, i.e. Google, and retrieve the documents in the top 3 pages returned by Google. As each raw document is in the HTML format, we parse the HTML page to obtain clean version of report articles with the following pre-processing procedure: (1) Discarding the invisible content by their tags, like <script> and <img>; and (2) Removing URLs and non-alphabetic characters from the visible content. Note that we do not remove stopwords in such a pre-processing procedure, because they provide important context information for our document embedding module in later stages.

Snippet candidate generation. We then generate snippet candidates for both the cleaned HTML contents and the review articles from the fact database (we use *Snopes.com* in our system). In particular, we first segment the documents into sentences. After segmentation, we then apply a greedy algorithm to generate snippet candidates by removing the sentences containing less than K words and grouping every L sentences into a snippet. Here, K and L are hyper-parameters that control both the accuracy and the efficiency of our model. There is a tradeoff for selecting K and L: (1) For effectiveness purposes, a too large K will ignore the relevant sentences, and a too large L will make a snippet contain irrelevant sentences, dampening the sentences that strongly represent the opinions; (2) For efficiency purposes, a too small K will include some meaningless short sentences, and a too small L will generate a large number of snippet candidates, which becomes the bottleneck of find relevant snippets in the online part, because embedding inference and similarity calculation will be applied to every candidate snippet when turning the related snippets to a claim. Through empirical studies, we set $L = 3$ and $K = 3$ to achieve a tradeoff between effectiveness and efficiency.

Relevant snippets extraction. To select relevant snippets from the candidates, previous work [8] discovers relevant snippets by calculating the overlapping unigrams and bigrams between the fact and candidate snippets. This method tends to miss the relevant snippets when snippets are expressed in different forms. To alleviate the issue, we propose to leverage document embedding techniques, which learns vector representations for both claims and snippets. Our document embedding module is based on Doc2vec [4], which extends word2vec [6] by treating the document as an inherent context, and combining it with the context in a sliding window to predict keywords. By learning the embeddings for claims and snippets, we calculate the cosine similarity between the vectors of the claim and each candidate snippet, and keep the snippets whose cosine similarity is larger than a threshold δ. With a validation set, we set $\delta = 0.55$ as it yields high-quality relevant snippets in practice.

In our ClaimVerif system, we have created a corpus containing 8367 related snippets from the editorial review articles of our fact database for our stance classifier, which will be introduced shortly. We also obtain a corpus containing 3.1M relevant snippets from Google report articles indexed by each claim, which will be used together with the fact database for article classification. Meanwhile, for online phase, the same relevant snippets generation process is deployed. The embedding vector of real-time queries are inferred using the trained embedding vectors. Since inferring the embedding vectors is another bottleneck in real-time usage, empirically we find setting the number of iteration to 150 achieves a good balance between accuracy and efficiency.

2.2 Two-Step Training

In the two-step offline training process, we first train a binary classifier to judge the truthfulness of stances, and then leverage the outputs of the stance classifier as input features to train an article classifier.

Stance Classification. We train the stance classifier based on the snippets extracted from the review articles in the fact database. Note that we do not include the snippets in the report articles acquired from the search engine. It is because the retrieved articles may not be the exact articles justifying the claim, and incorporating them may leads to unwanted noise in the early stage. The key rationale behind this strategy is to train a high-quality classifier to reveal the support or refute opinion of snippets.

We train the stance classifier using Random Forest, where the features are the embeddings of snippets selected from the review articles of the claims, and the training labels are the true/false labels of the claims. The classifier outputs two scores for each snippet, namely the score of snippet supporting the claim, and the score of snippet refuting the claim. We use these two scores as input features for the next step of article classification.

Article Classification. The article classification component is designed to judge whether an article is supporting or refuting a claim. To train the article classifier, we combine the snippets from the fact database and the search engine as the input features, and use the outputs of the stance classifier as feature scores. Moreover, to obtain reliable features, we use the top three stances with the highest absolute scores for each article, and compute the average of supporting/reputing stance scores as two additional features. We find that such a feature selection strategy leads to discriminative features for delineating the three different cases of an article: fully supportive, completely refuting, and a mixture of being supportive and refuting.

In addition to the above features, the way a stance is stated is also important to understanding the strength of the opinions. If a stance is expressed in an objective or unbiased style, we may rate them as stances with high confidence. On the contrary, if a stance is stated in a subjective style, its score would be lowered for less confidence. To capture such an observation, we resort to the following linguistic features [9]: (1) Factive verbs that presuppose the truth of their complement clause, e.g. know, realize, regret, forget, find out, discover, learn, note, notice; (2) Assertive verbs that provide the certainty for the proposition holds, e.g. think, believe, suppose, expect, imagine, guess, seem, appear; (3) Mitigating words, e.g. about, almost, apparent, apparently; (4) Report verbs, e.g. accuse, acknowledge, add, admit, advise, agree, alert, allege; (5) Discourse makers, e.g. could, maybe; and (6) Subjective/bias, e.g. abuse, accept, account. In our system implementation, we encode them into one-hot feature vectors for training the Random Forest classifier. After the training process, the output is used as the supporting/refuting opinion score of an article to a query claim.

2.3 Claim Credibility Assessment

The core of the credibility assessment component is underpinned by a source quality assessment algorithm, followed by a score reweighing function.

Source Quality Assessment Previous work [7] on source quality assessment have used PageRank and AlexaRank to measure the reliability of websites. However, such measures only indicate source popularities without quantifying source credibilities. Another recent work [8] computes the credibilities of websites based on the proportions of the relevant stances whose opinions are coincident with the ground truth of the claims. To address the above problems, we use Web of Trust (WOT)[2], a service that calculates website reputations using ratings from users and evaluations from third-party sources. When the trustworthiness lower bound retrieved from WOT is less than 10, we judge the website as not credible, and degrade its respective stance scores in the next score reweighing step. In this way, we penalize the stances provided by highly untrustworthy sources.

Score Reweighing We are now ready to describe the score reweighing function. We evaluate the credibility of a user-given claim by combining the stance score and the credibility of source websites. In particular, we re-scale the score from WOT to $[0, 1]$, and the compute the predicted label of claim i y_i by Eq.1:

$$y_i = \underset{a \in \{T,F\}}{\arg\max} \sum_j (reliability_j * Article(y_{ij} = a)), \quad (1)$$

where y_{ij} is the judgment of claim i from article of source j. We have $reliability_j = score_{WOT}$ when the score of source j from WOT is less than 10, otherwise $reliability_j = 1$. The credibility of claim i is the weighted sum of stance scores of the label y_i normalized by the sum of the weighted support and refute stance scores.

3 DEMONSTRATION

Fig. 2 illustrates two example claims and the results returned by ClaimVerif. With our system, the users input a claim as a query, and the system will return: (1) the top-related evidences with sources; (2) the true/fake judgment with confidence score; and (3) the time used for verifying the claim. In practice, the overall running time of processing one user query takes 10 to 80 seconds — depending on the number of report articles retrieved from the search engine and the length of the articles.

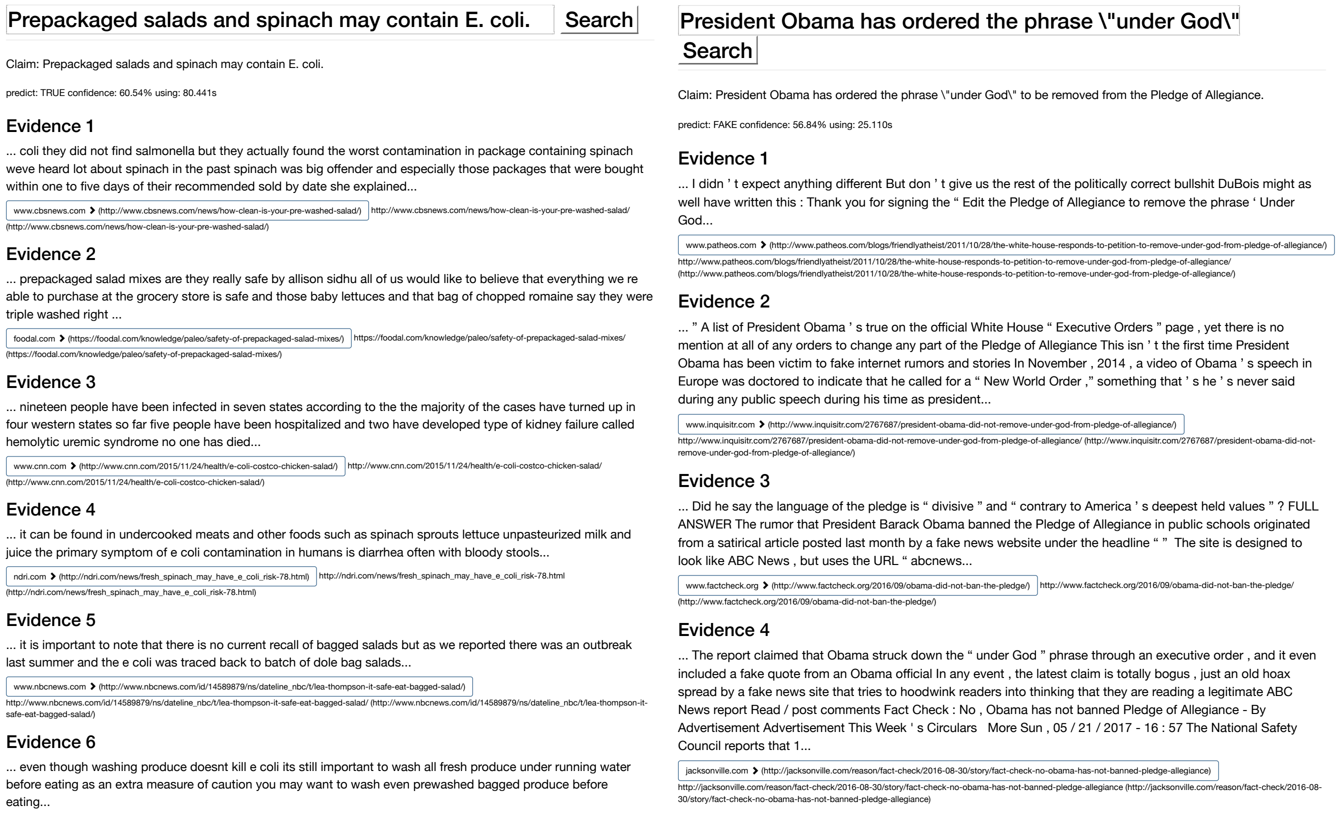
We will also demonstrate the comparative results of our system and an existing method [8]. First, for quantitative evaluation, we have used a set of claims and their ground-truth labels. The test data set consists of 105 claims, which are excluded from the fact database in the training stage. With the ground-truth test set, we compare the performance of our system with [8], a method based on distant supervision. Table 1 shows the accuracy of the two-step training. The accuracy of our stance classifier improves the baseline method by 7% out of the whole testing set. The baseline method uses the words as the feature trained by a linear classifier, which may not be accurate in the stance classification. Instead, we train the distributed representation of claims and snippets in the training set, and infer the vector representations of claims and snippets in the test set. With this method, we can detect relevant snippets with higher precisions and recalls.

	Stance Classifier	Claim Credibility
Baseline	76.69%	81.39%
ClaimVerif	83.62%	85.25%

Table 1: Accuracy comparison for the stance classification and claim credibility assessment components.

As for claim credibility assessment, the baseline method uses the learned stance score to calculate the source quality. However, in our experiments we find that that method amplifies the errors made by stance classifier and propagates the errors to the claim credibility justification stage. By incorporating the source credibility evaluation service into our system, we can effectively blacklist stances from untrustworthy websites. As a concrete example, for the claim “President Obama has ordered the phrase ‘under God’ to be removed from the Pledge of Alliance”, the top-ranked evidences without the source quality module are provided by *abcnews.com.co*,

[2] https://www.mywot.com/wiki/API



Prepackaged salads and spinach may contain E. coli.

Claim: Prepackaged salads and spinach may contain E. coli.

predict: TRUE confidence: 60.54% using: 80.441s

Evidence 1

... coli they did not find salmonella but they actually found the worst contamination in package containing spinach weve heard lot about spinach in the past spinach was big offender and especially those packages that were bought within one to five days of their recommended sold by date she explained...

www.cbsnews.com ❯ (http://www.cbsnews.com/news/how-clean-is-your-pre-washed-salad/) http://www.cbsnews.com/news/how-clean-is-your-pre-washed-salad/ (http://www.cbsnews.com/news/how-clean-is-your-pre-washed-salad/)

Evidence 2

... prepackaged salad mixes are they really safe by allison sidhu all of us would like to believe that everything we re able to purchase at the grocery store is safe and those baby lettuces and that bag of chopped romaine say they were triple washed right ...

foodal.com ❯ (https://foodal.com/knowledge/paleo/safety-of-prepackaged-salad-mixes/) https://foodal.com/knowledge/paleo/safety-of-prepackaged-salad-mixes/ (https://foodal.com/knowledge/paleo/safety-of-prepackaged-salad-mixes/)

Evidence 3

... nineteen people have been infected in seven states according to the the majority of the cases have turned up in four western states so far five people have been hospitalized and two have developed type of kidney failure called hemolytic uremic syndrome no one has died...

www.cnn.com ❯ (http://www.cnn.com/2015/11/24/health/e-coli-costco-chicken-salad/) http://www.cnn.com/2015/11/24/health/e-coli-costco-chicken-salad/ (http://www.cnn.com/2015/11/24/health/e-coli-costco-chicken-salad/)

Evidence 4

... it can be found in undercooked meats and other foods such as spinach sprouts lettuce unpasteurized milk and juice the primary symptom of e coli contamination in humans is diarrhea often with bloody stools...

ndri.com ❯ (http://ndri.com/news/fresh_spinach_may_have_e_coli_risk-78.html) http://ndri.com/news/fresh_spinach_may_have_e_coli_risk-78.html (http://ndri.com/news/fresh_spinach_may_have_e_coli_risk-78.html)

Evidence 5

... it is important to note that there is no current recall of bagged salads but as we reported there was an outbreak last summer and the e coli was traced back to batch of dole bag salads...

www.nbcnews.com ❯ (http://www.nbcnews.com/id/14589879/ns/dateline_nbc/t/lea-thompson-it-safe-eat-bagged-salad/) http://www.nbcnews.com/id/14589879/ns/dateline_nbc/t/lea-thompson-it-safe-eat-bagged-salad/ (http://www.nbcnews.com/id/14589879/ns/dateline_nbc/t/lea-thompson-it-safe-eat-bagged-salad/)

Evidence 6

... even though washing produce doesnt kill e coli its still important to wash all fresh produce under running water before eating as an extra measure of caution you may want to wash even prewashed bagged produce before eating...

(a) A claim judged as true.

President Obama has ordered the phrase \"under God\"

Claim: President Obama has ordered the phrase \"under God\" to be removed from the Pledge of Allegiance.

predict: FAKE confidence: 56.84% using: 25.110s

Evidence 1

... I didn ’ t expect anything different But don ’ t give us the rest of the politically correct bullshit DuBois might as well have written this : Thank you for signing the “ Edit the Pledge of Allegiance to remove the phrase ‘ Under God...

www.patheos.com ❯ (http://www.patheos.com/blogs/friendlyatheist/2011/10/28/the-white-house-responds-to-petition-to-remove-under-god-from-pledge-of-allegiance/) http://www.patheos.com/blogs/friendlyatheist/2011/10/28/the-white-house-responds-to-petition-to-remove-under-god-from-pledge-of-allegiance/ (http://www.patheos.com/blogs/friendlyatheist/2011/10/28/the-white-house-responds-to-petition-to-remove-under-god-from-pledge-of-allegiance/)

Evidence 2

... ” A list of President Obama ’ s true on the official White House “ Executive Orders ” page , yet there is no mention at all of any orders to change any part of the Pledge of Allegiance This isn ’ t the first time President Obama has been victim to fake internet rumors and stories In November , 2014 , a video of Obama ’ s speech in Europe was doctored to indicate that he called for a “ New World Order ,” something that ’ s he ’ s never said during any public speech during his time as president...

www.inquisitr.com ❯ (http://www.inquisitr.com/2767687/president-obama-did-not-remove-under-god-from-pledge-of-allegiance/) http://www.inquisitr.com/2767687/president-obama-did-not-remove-under-god-from-pledge-of-allegiance/ (http://www.inquisitr.com/2767687/president-obama-did-not-remove-under-god-from-pledge-of-allegiance/)

Evidence 3

... Did he say the language of the pledge is “ divisive ” and “ contrary to America ’ s deepest held values ” ? FULL ANSWER The rumor that President Barack Obama banned the Pledge of Allegiance in public schools originated from a satirical article posted last month by a fake news website under the headline “ ” The site is designed to look like ABC News , but uses the URL “ abcnews...

www.factcheck.org ❯ (http://www.factcheck.org/2016/09/obama-did-not-ban-the-pledge/) http://www.factcheck.org/2016/09/obama-did-not-ban-the-pledge/ (http://www.factcheck.org/2016/09/obama-did-not-ban-the-pledge/)

Evidence 4

... The report claimed that Obama struck down the “ under God ” phrase through an executive order , and it even included a fake quote from an Obama official In any event , the latest claim is totally bogus , just an old hoax spread by a fake news site that tries to hoodwink readers into thinking that they are reading a legitimate ABC News report Read / post comments Fact Check : No , Obama has not banned Pledge of Allegiance - By Advertisement Advertisement This Week ' s Circulars More Sun , 05 / 21 / 2017 - 16 : 57 The National Safety Council reports that 1...

jacksonville.com ❯ (http://jacksonville.com/reason/fact-check/2016-08-30/story/fact-check-no-obama-has-not-banned-pledge-allegiance) http://jacksonville.com/reason/fact-check/2016-08-30/story/fact-check-no-obama-has-not-banned-pledge-allegiance (http://jacksonville.com/reason/fact-check/2016-08-30/story/fact-check-no-obama-has-not-banned-pledge-allegiance)

(b) A claim judged as fake.

Figure 2: Two claims judged by ClaimVerif

which is marked as a website frequently spreading rumors. By lowering the score of it, our ClaimVerif system can exclude these untrustworthy sources, as shown in Fig. 2b.

4 CONCLUSION

We have presented ClaimVerif, a novel real-time claim verification system that provides credibility assessment for a user-given query claim and explains the assessment with supporting evidences. Unlike existing tools, our system captures the semantic similarity between the candidate snippets and the claim by learning the distributed representations of them. Leveraging data from fact databases and reports from the Web, our system performs a two-step training to obtain a high-quality stance classifier and a reliable article classifier. Furthermore, it incorporates a source quality service to degrade the score of stances from untrustworthy sources, alleviating the negative effects from rumor spreaders. Our empirical evaluation has shown that our system returns quality results for a wide variety of query claims.

5 ACKNOWLEDGEMENT

Research was sponsored in part by the U.S. Army Research Lab. under Cooperative Agreement No. W911NF-09-2-0053 (NSCTA), National Science Foundation IIS 16-18481 and NSF IIS 17-04532, and grant 1U54GM114838 awarded by NIGMS through funds provided by the trans-NIH Big Data to Knowledge (BD2K) initiative.

REFERENCES

[1] Hunt Allcott and Matthew Gentzkow. 2017. *Social media and fake news in the 2016 election.* Technical Report. National Bureau of Economic Research.

[2] Sardar Hamidiain and Mona Diab. 2015. Rumor detection and classification for twitter data. In *SOTICS, IARIA*.

[3] Zhiwei Jin, Juan Cao, Yu-Gang Jiang, and Yongdong Zhang. 2014. News credibility evaluation on microblog with a hierarchical propagation model. In *Data Mining (ICDM), 2014 IEEE International Conference on*. IEEE, 230–239.

[4] Quoc V Le and Tomas Mikolov. 2014. Distributed Representations of Sentences and Documents.. In *ICML*, Vol. 14. 1188–1196.

[5] Jing Ma, Wei Gao, Zhongyu Wei, Yueming Lu, and Kam-Fai Wong. 2015. Detect rumors using time series of social context information on microblogging websites. In *CIKM*. ACM, 1751–1754.

[6] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*.

[7] Kashyap Popat, Subhabrata Mukherjee, Jannik Strötgen, and Gerhard Weikum. 2016. Credibility Assessment of Textual Claims on the Web. In *CIKM*. ACM.

[8] Kashyap Popat, Subhabrata Mukherjee, Jannik Strötgen, and Gerhard Weikum. 2017. Where the Truth Lies: Explaining the Credibility of Emerging Claims on the Web and Social Media. In *WWW*.

[9] Marta Recasens, Cristian Danescu-Niculescu-Mizil, and Dan Jurafsky. 2013. Linguistic Models for Analyzing and Detecting Biased Language.. In *ACL*.

[10] Fan Yang, Yang Liu, Xiaohui Yu, and Min Yang. 2012. Automatic detection of rumor on Sina Weibo. In *SIGKDD Workshop on Mining Data Semantics*. ACM, 13.