Toward Automated Fact-Checking: Detecting Check-worthy Factual Claims by ClaimBuster

Naeemul Hassan*
Department of Computer and Information Science, University of Mississippi

Fatma Arslan
Department of Computer Science and Engineering, University of Texas at Arlington

Chengkai Li
Department of Computer Science and Engineering, University of Texas at Arlington

Mark Tremayne
Department of Communication, University of Texas at Arlington

ABSTRACT

This paper introduces how ClaimBuster, a fact-checking platform, uses natural language processing and supervised learning to detect important factual claims in political discourses. The claim spotting model is built using a human-labeled dataset of check-worthy factual claims from the U.S. general election debate transcripts. The paper explains the architecture and the components of the system and the evaluation of the model. It presents a case study of how ClaimBuster live covers the 2016 U.S. presidential election debates and monitors social media and Australian Hansard for factual claims. It also describes the current status and the long-term goals of ClaimBuster as we keep developing and expanding it.

1 INTRODUCTION

This paper introduces ClaimBuster, an ongoing project toward automated fact-checking. The focus is on explaining the claim spotting component of the system which discovers factual claims that are worth checking from political discourses. This component has been deployed and substantially tested in real-world use cases. We also present the current prototype and the goals regarding the system's other components. While the project has been described in a few short papers and non-archival publications [11–13], this paper provides a detailed and holistic account of the system for the first time.

Our society is struggling with unprecedented amount of falsehoods, hyperboles and half-truths which do harm to wealth, democracy, health, and national security. People and organizations make claims about "facts" all the time. Politicians repeatedly make the same false claims. [1] Fake news floods the cyberspace and even allegedly influenced the 2016 election. [2] In fighting false information, the number of active fact-checking organizations has grown from 44 in 2014 to almost 100 in 2016. [3] Fact-checkers vet claims by investigating relevant data and documents and publish their verdicts. For instance, PolitiFact.com, one of the earliest and most popular fact-checking projects, gives factual claims truthfulness ratings such as True, Mostly True, Half true, Mostly False, False, and even "Pants on Fire". In the U.S., the election year made fact-checking a household terminology. For example, during the first presidential debate on September 26, 2016, NPR.org's live fact-checking website drew 7.4 million pageviews and delivered its biggest traffic day ever.

The challenge is that the human fact-checkers cannot keep up with the amount of misinformation and the speed at which they spread. The reason is that fact-checking is an intellectually demanding, laborious and time-consuming process. It takes about one day to research and write a typical article about a factual claim [11]. (By contrast, Leskovec, Backstrom and Kleinberg [14] found a meme typically moves from the news media to blogs in just 2.5 hours.) These difficulties, exacerbated by a lack of resources for investigative journalism, leaves many harmful claims unchecked, particularly at the local level.

This challenge creates an opportunity for automated fact-checking systems. On the other hand, fact-checking technology is clearly falling behind, as there is simply no existing system that truly does automated fact-checking. Today's professional fact-checkers diligently perform their work as an art, following the good practices in data journalism [9] and investigative journalism [5]. A recent white paper [8] surveys existing tools that can be put together. While the relevant tools and techniques can assist fact-checking in various steps, here and there, a full-fledged, end-to-end solution does not exist. There have been a few attempts [26], [4] but the efforts did not lead to such fact-checking systems.

Starting from December 2014, we have been building ClaimBuster, an end-to-end system for computer-assisted fact-checking. ClaimBuster uses machine learning, natural language processing, and database query techniques to aid fact-checking. It monitors live discourses (e.g., interviews, speeches and debates), social media, and news to catch factual claims, detects matches with a curated repository of fact-checks from professionals, and delivers the matches instantly to readers and viewers (e.g., by displaying a pop-up warning if a presidential candidate makes a false claim during a live

*Work performed while at the University of Texas at Arlington.
[1] A. D. Holan. *All Politicians Lie. Some Lie More Than Others.* The New York Times, December 11, 2015. http://goo.gl/Js0XGg
[2] Hannah Jane Parkinson. *Click and elect: how fake news helped Donald Trump win a real election.* The Guardian, November 14, 2016. https://goo.gl/Of6nw7
[3] http://reporterslab.org/global-fact-checking-up-50-percent/
[4] T. Wilner. *Fail and move on: Lessons from automated fact-checking experiments.* Poynter, September 7, 2016. http://goo.gl/G0l54Y

KDD '17, August 13-17, 2017, Halifax, NS, Canada

ACM ISBN 978-1-4503-4887-4/17/08…$15.00
https://doi.org/10.1145/3097983.3098131

debate). For various types of new claims not checked before, ClaimBuster automatically translates them into queries against knowledge databases and reports whether they check out. For claims where humans must be brought into the loop, ClaimBuster provides algorithmic and computational tools to assist lay persons and professionals in understanding and verifying the claims. Its use will be expanded to verify both political and non-political claims in many types of narratives, discourses and documents such as sports news, legal documents, and financial reports.

ClaimBuster already produces true-or-false verdicts for certain types of factual claims. The development of the full-fledged system is still ongoing. A description of its current status is in Section 5 and a demonstration is presented at [10]. In this paper, we focus on a key component of the system, claim spotting, which discovers factual claims that are worth checking. Given the plethora of discourses and narratives we are constantly exposed to, ClaimBuster gives each sentence a score that indicates how likely it contains an important factual claim that should be checked. This essentially provides a priority ranking that helps fact-checkers efficiently focus on the top-ranked sentences without painstakingly sifting through a large number of sentences. ClaimBuster's scorer was tested in real-time during the live coverage of all primary election and general election debates for the 2016 election. Closed captions of the debates on live TV broadcasts, captured by a decoding device, were fed to ClaimBuster, which immediately scored each sentence spoken by the candidates and posted top-scored claims to the project's website (idir.uta.edu/claimbuster) and Twitter account (@ClaimBusterTM). Post-hoc analysis of the claims checked by professional fact-checkers at CNN, PolitiFact.com and FactCheck.org reveals a highly positive correlation between ClaimBuster and journalism organizations in deciding which claims to check. ClaimBuster has also been continuously monitoring Twitter and retweeting the check-worthy factual claims it finds in people's tweets (see https://twitter.com/ClaimBusterTM). Recently it also started to monitor "Hansard" [5] – the transcripts of proceedings of the Australian parliament (idir.uta.edu/claimbuster/hansard).

The project has received wide recognition in the fact-checking community and substantial media coverage. [6] [7] [8] [9] [10] [11] The white paper [8] calls ClaimBuster a tool with "the most advanced generalised automatic claim spotting." [8] Others considered it "perhaps the biggest development to date" in ranking claims [12] and "a pretty useful guide for journalists and those members of the public who wish to spend time using an algorithm to help find facts." [13]

ClaimBuster, upon completion, is positioned to become the first-ever end-to-end fact-checking system. It can benefit a large base of potential users. It directly benefits citizens and consumers by improving information accuracy and transparency. It helps news organizations speed their fact-checking process and also ensure the accuracy of their own news stories. Businesses can use ClaimBuster to identify falsehoods in their competitors' and their own reports and press releases. It can also assist professionals such as lawyers in verifying documents.

The organization of this paper is as follows. Section 2 discusses related work. Section 3 formulates the claim spotting problem as a classification and ranking task, explains the solution in ClaimBuster, and presents the evaluation results. Section 4 describes a case study– how ClaimBuster was used to cover the 2016 U.S. presidential election. Section 5 describes the architecture of ClaimBuster and the current status of other components of the system.

2 RELATED WORK

In the last few years, several projects and startup companies attempted at building computer-assisted fact-checking systems [26]. Trooclick (http://storyzy.com/) aimed at fact-checking financial news by comparing IPO stories against SEC (Securities and Exchange Commission) filings. TruthTeller, [14] a project by Washington Post, detects claims in the speeches on TV and matches them against the fact-checks from Washington Post and other organizations. LazyTruth (http://www.lazytruth.com/) aimed at finding false claims in email chain letters. A few other projects take the crowdsourcing approach to fact-checking. Fiskkit (http://fiskkit.com/) operates a platform which allows users to break apart a news article and discuss the accuracy of its content piece by piece. The Grasswire project (https://www.grasswire.com/about-us/) is, to some extent, the Wikipedia counterpart of news website. Their contributors, who are not necessarily journalists, collaborate in a Slack(https://slack.com/) channel in which they pitch, source, verity, write, and edit news stories. Truth Goggles, [15] initially aimed at automated fact-checking, is a tool that allows users to annotate web content for fact-checking.

Since most of these projects were never past proof-of-concept stage before they ceased operations, there is only limited available information. Nevertheless, it appears none of the projects takes the structured approach of ClaimBuster to model and understand factual claims themselves. None does algorithmic fact-checking. Truth Teller resembles the claim matching component of ClaimBuster, but it resorts to straightforward string matching instead of understanding the structure and semantics of claims. None of the projects developed the capability of spotting and ranking claims based on their check-worthiness.

There are several lines of academic research which are related to fact-checking. Vlachos and Riedel [24] analyzed the tasks in fact

[5] www.aph.gov.au/Parliamentary_Business/Hansard

[6] G. Pogrund. *Post-truth v tech: could machines help us call out politicians' and journalists' lies?* newstatesman.com, August 17, 2016. http://goo.gl/eGf5DP

[7] C. Albeanu. *What would an automated future look like for verification in the newsroom?* journalism.co.uk, April 8, 2016. http://goo.gl/KKPgnK

[8] T. Walk-Morris. *The Future of Political Fact-Checking*, NiemanReports, March 23, 2016. http://goo.gl/syUdjv

[9] B. Mullin. *Knight Foundation backs 20 media projects with Prototype Fund.* Poynter, November 3, 2015. http://goo.gl/HsjJXq

[10] C. Silverman. *In search of fact checking's 'Holy Grail': News outlets might not get there alone.* First Draft, October 30, 2015. http://goo.gl/KFxBSz

[11] G. Selby. *Sifting balderdash from truth gets a boost from computers.* Austin American-Statesman, August 8, 2015. http://goo.gl/FCzY3c

[12] K. Moreland and B. Doerrfeld. *Automated Fact Checking: The Holy Grail of Political Communication.* Nordic APIs, February 25, 2016. http://goo.gl/uhsnyT

[13] P. Fray. *Is that a fact? Checking politicians' statements just got a whole lot easier.* The Guardian, April 18, 2016. http://goo.gl/1UJfzU

[14] https://www.washingtonpost.com/news/ask-the-post/wp/2013/09/25/announcing-truth-teller-beta-a-better-way-to-watch-political-speech/

[15] http://www.poynter.org/2014/truth-goggles-launches-as-an-annotation-tool-for-journalists/256917/

checking and presented a dataset of factual claims collected from *PolitiFact.com* and *Channel4.com*. *Rumor detection* aims at finding rumors in social media, by considering linguistic signals in the content of social media posts, signature text phrases in users' comments that express skepticism, how they were spread, as well as the credibility of the authors based on track record [3, 7, 19]. They do not resort to structured analysis of claims themselves. *Truth discovery* is concerned about the specific problem of detecting true facts from conflicting data sources [15]. They do not directly consider factual claims. Instead, they assume the input of a collection of (contradicting) tuples that record the property values of objects. Ciampaglia et al. [4] proposed a method for fact-checking using knowledge graphs by finding the shortest path between entity nodes. Shi et al. [22] mine knowledge graphs to find missing links between entities. This approach, though more related to the general problem of *link prediction* [6, 16] than fact-checking, can potentially identify supporting evidence for facts that are not recorded in knowledge graphs. We note that none of the aforementioned works on truth discovery, link prediction, and fact-checking using knowledge graphs aims at an end-to-end system, as they do not directly cope with factual claims.

To the best of our knowledge, no prior study has focused on computational methods for detecting factual claims and discerning their importance. The most relevant line of work is subjectivity analysis of text (e.g., [2, 25, 27]) which classifies sentences into objective and subjective categories. However, not all objective sentences are check-worthy important factual claims. In Section 3.5, we present a comparison between subjectivity analysis and ClaimBuster of which the results demonstrate the inability of subjectivity identifiers in discerning factual claims.

3 CLAIM SPOTTING: CHECK-WORTHY FACTUAL CLAIMS DETECTION

We model the claim spotting problem as a classification and ranking task and we follow a supervised learning approach to address it. We constructed a labeled dataset of spoken sentences by presidential candidates during past presidential debates. Each sentence is given one of three possible labels–it is not a factual claim; it is an unimportant factual claim; it is an important factual claim. We trained and tested several multi-class classification models using the labeled dataset. Experiment results demonstrated the promising accuracy of the models. We further compared our model with existing subjectivity classifiers and demonstrated that subjectivity identifiers are incapable of discerning factual claims.

3.1 Classification and Ranking

We categorize the sentences spoken in the presidential debates into three categories:

Non-Factual Sentence (NFS): Subjective sentences (opinions, beliefs, declarations) and many questions fall under this category. These sentences do not contain any factual claim. Below are two such examples.

- *But I think it's time to talk about the future.*
- *You remember the last time you said that?*

Unimportant Factual Sentence (UFS): These are factual claims but not check-worthy. The general public will not be interested in knowing whether these sentences are true or false. Fact-checkers do not find these sentences as important for checking. Some examples are as follows.

- *Next Tuesday is Election day.*
- *Two days ago we ate lunch at a restaurant.*

Check-worthy Factual Sentence (CFS): They contain factual claims and the general public will be interested in knowing whether the claims are true. Journalists look for these type of claims for fact-checking. Some examples are:

- *He voted against the first Gulf War.*
- *Over a million and a quarter Americans are HIV-positive.*

Given a sentence, the objective of ClaimBuster's claim spotting is to derive a score that reflects the degree by which the sentence belongs to CFS. Many widely-used classification methods support ranking naturally. For instance, consider Support Vector Machine (SVM). We treat CFSs as positive instances and both NFSs and UFSs as negative instances. SVM finds a decision boundary between the two types of training instances. Following Platt's scaling technique [18], for a given sentence x to be classified, we calculate the posterior probability of the sentence belonging to CFS using SVM's decision function. The sentences are then ranked by their probability scores:

$$score(x) = P(class = \text{CFS}|x)$$

3.2 Data Labeling

We need to collect a labeled dataset which, for each sentence from the U.S. general election presidential debates, indicates its label among the three options [NFS, UFS, CFS]. Such a dataset does not exist. Our dataset, once completed and released, will be a valuable asset to the research community and practitioners.

Dataset The custom of organizing debates between U.S. presidential candidates before a general election started in 1960. There has been a total of 15 presidential elections from 1960 to 2012. Except 1964, 1968, and 1972 there have been debates before all the 12 remaining elections. The number of debates before an election varies from year to year; for example, there were two and three debates before 1988 and 2012 elections, respectively. We have collected the transcripts of all the debates occurred during 1960–2012. In total, there are 30 debates in these 11 election years. There are 28029 sentences in these transcripts. Using parsing rules and human annotation, we identified the speaker of each sentence. 23075 sentences are spoken by the presidential candidates and 4815 by the debate moderators. We concentrated on the 20788 sentences spoken by the candidates which are at least 5 words long.

Ground-truth Collection Website We developed a rich and controlled data collection website (http://idir-server2.uta.edu/classifyfact_survey) to collect the ground-truth labels of the sentences. Figure 1 shows its interface. A participant is presented one sentence at a time. The sentence is randomly selected from the set of sentences not seen by the participant before. They can assign one of three possible labels [NFS, UFS, CFS] for the sentence. If the participant is not confident to assign a label for a sentence, they can skip it. It is also possible to go back and modify previous responses.

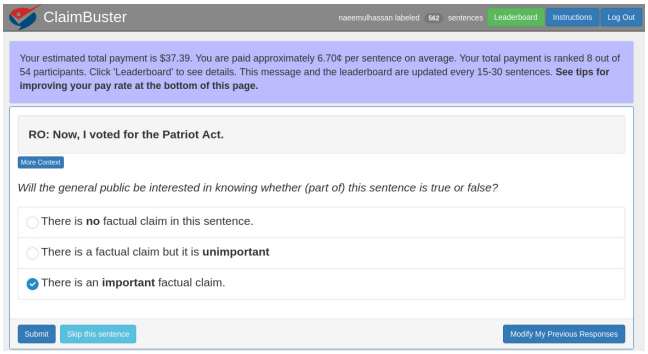

Figure 1: Data collection interface

With just the text of a sentence itself, it is sometimes difficult to determine its label. The interface has a "more context" button. When it is clicked, the system shows the four preceding sentences of the sentence in question which may help the participant understand its context. We observe that, about 14% of the time, participants chose to read the context before labeling a sentence.

Participant Recruitment and Training We recruited paid participants (mostly university students, professors and journalists who are aware of U.S. politics) using flyers, social media, and direct emails. We use 30 selected sentences to train all the participants. Every participant must go through all these 30 sentences at the very beginning. After they label a sentence, the website will immediately disclose its ground-truth label and explain it. Furthermore, we arranged multiple on-site training workshops for participants that were available. During each workshop, at least two experts were present to clear the doubts the participants may have about the data collection website and process. Through interviews with the participants, we observed that these training measures were important in helping the participants achieve high work quality.

We chose to not use crowdsourcing platforms such as Amazon Mechanical Turk and CrowdFlower, due to the complex nature of the task and its requirement for a participant to have basic knowledge about U.S. politics. We will not be able to run the on-site training workshops for the participants on such platforms. We indeed performed a pilot run on CrowdFlower with a small dataset and we were not impressed by the quality of the collected data. It will be interesting to conduct a thorough comparison with the data collection approach of using such a platform for our data.

Quality Assurance To detect spammers and low-quality participants, we selected 1,032 (731 NFS, 63 UFS, 238 CFS) sentences from all the sentences for screening purpose. Three experts agreed upon the labels of these sentences. On average, one out of every ten sentences given to a participant (without letting the participant know) was randomly chosen to be a screening sentence. First, a random number decides the type (NFS, UFS, CFS) of the sentence. Then, the screening sentence is randomly picked from the pool of screening sentences of that particular type. The degree of agreement on screening sentences between a participant and the three experts is one of the factors in measuring the quality of the participant. For a screening sentence, when a participant's label matches the experts' label, s/he is rewarded with some points. If it does not match, s/he is penalized. We observe that not all kinds of mislabeling has equal significance. For example, labeling an NFS sentence as a CFS is a more critical mistake than labeling a UFS as a CFS. We defined weights for different types of mistakes and incorporated them into the quality measure.

Formally, given $SS(p)$ as the set of screening sentences labeled by a participant p, the labeling quality of p (LQ_p) is

$$LQ_p = \frac{\sum_{s \in SS(p)} \gamma^{lt}}{|SS(p)|}$$

where γ^{lt} is the weight factor when p labeled the screening sentence s as l and the experts labeled it as t. Both $l, t \in \{NFS, UFS, CFS\}$. We set $\gamma^{lt} = -0.2$ where $l = t$, $\gamma^{lt} = 2.5$ where $(l, t) \in \{(NFS, CFS), (CFS, NFS)\}$ and $\gamma^{lt} = 0.7$ for all other combinations. The weights are set empirically. If $LQ_p \leq 0$ for a participant p, we designate p as a top-quality participant. A total of 374 participants contributed in the data collection process so far. Among them, 86 are top-quality participants. Figure 2 shows the frequency distribution of LQ_p for all participants.

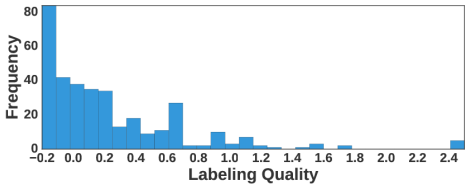

Figure 2: Frequency distribution of participants' labeling quality

Incentives We devised a monetary reward program to encourage the participants to perform high-quality labeling. A participant p's payment depends on their pay rate per sentence R_p (in cents) and their number of labeled sentences. R_p depends on LQ_p, the lengths of the labelled sentences, and the percentage of skipped sentences. The reason behind the later two factors is to discourage participants from skipping longer and more challenging sentences and to reward them for working on long, complex sentences. After multiple rounds of empirical analysis, we set R_p as

$$R_p = \frac{L_p}{L}^{1.5} * (3 - \frac{7 * LQ_p}{0.2}) * 0.6^{\frac{|SKIP_p|}{|ANS_p|}}$$

where, L is the average length of all the sentences, L_p is the average length of sentences labeled by p, ANS_p is the set of sentences labeled by p and $SKIP_p$ is the set of sentences skipped by p. The numerical values in the above equation were set in such a way that it would be possible for a top-quality participant to earn up to 10 cents for each sentence.

The data-collection website also features a leaderboard which allows participants to see their rank positions by pay rate and total payment. This is designed to encourage serious participants to perform better and discourage spammers from further participation. Along with the leaderboard, the website provides helpful tips and messages from time to time to keep the participants motivated.

Stopping Condition

A sentence s will not be selected for further labeling if for $X \in \{NFS, UFS, CFS\}$, $\exists X$ such that $s_X \geq 2 \wedge s_X > (s_{NFS} + s_{UFS} + s_{CFS})/2$ where, s_X denotes the number of top-quality labels of type X assigned to s.

This condition ensures that a sentence has received a reasonable number of labels from top-quality participants and the majority of

them agreed on a particular label. We assign the majority label as the ground-truth of that sentence.

The data collection continued for about 20 months in multiple phases. We collected 76, 552 labels among which 52, 333 (68%) are from top-quality participants. There are 20, 617 (99.17%) sentences which satisfy the above stopping condition. Table 1 shows the distribution of the classes in these sentences. Note that we perform all the experiments presented in this paper using a set of 8, 231 sentences which were labeled at the earlier stage of the data collection process. However, in Figure 4, we present the effects of different dataset sizes on the performance of the models. Further details are provided in the Section 3.4.

Table 1: Class distribution

	Count	Percentage
NFS	13671	66.31
UFS	2097	10.17
CFS	4849	23.52

3.3 Feature Extraction

We extracted multiple categories of features from the sentences. We use the following sentence to explain the features.

When President Bush came into office, we had a budget surplus and the national debt was a little over five trillion.

Sentiment: We used AlchemyAPI to calculate a sentiment score for each sentence. The score ranges from -1 (most negative sentiment) to 1 (most positive sentiment). The above sentence has a sentiment score -0.846376.

Length: This is the word count of a sentence. The natural language toolkit NLTK was used for tokenizing a sentence into words. The example sentence has length 21.

Word: We used words in sentences to build tf-idf features. After discarding stop-words and applying stemming, we had 6, 549 distinct tokens.

Part-of-Speech (POS) Tag: We applied the NLTK POS tagger on the sentences. There are 43 POS tags in the corpus. We constructed a feature for each tag. For a sentence, the count of words belonging to a POS tag is the value of the corresponding feature. In the example sentence, there are 3 words (came, had, was) with POS tag VBD (Verb, Past Tense) and 2 words (five, trillion) with POS tag CD (Cardinal Number).

Entity Type: We used AlchemyAPI to extract entities from sentences. There are 2, 727 entities in the labeled sentences. They belong to 26 types. The above sentence has an entity "Bush" of type "Person". We constructed a feature for each entity type. For a sentence, its number of entities of a particular type is the value of the corresponding feature.

Feature Selection: There are 6, 615 features in total. To identify the best discriminating features, we performed feature selection. We trained a random forest classifier for which we used GINI index to measure the importance of features in constructing each decision tree. The overall importance of a feature is its average importance over all the trees. Figure 3 shows the importance of the 30 best features in the forest. The black solid lines indicate the standard deviations of importance values. Category types are prefixes to feature names. The top features are quite intuitive. For instance, the most discriminating feature is the POS tag *VBD* which indicates the past form of a verb, which is often used to describe something happened in the past. The second most discriminating feature is the POS tag CD (Cardinal Number)–check-worthy factual claims are more likely to contain numeric values (45% of CFSs in our dataset) and non-factual sentences are less likely to contain numeric values (6% of NFSs in our dataset).

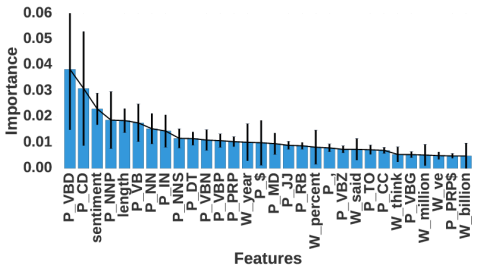

Figure 3: Feature importance

3.4 Evaluation

We performed 3-class (NFS/UFS/CFS) classification using several supervised learning methods, including Multinomial Naive Bayes Classifier (NBC), Support Vector Machine (SVM) and Random Forest Classifier (RFC). These methods were evaluated by 4-fold cross-validation. We experimented with various combinations of the extracted features. Table 2 shows the performances of the methods using combinations of words, POS tags, and entity type features. SVM had the best accuracy in general. On the CFS class, SVM combined with words, POS tags and entity types achieved 72% precision (i.e., it is accurate 72% of the time when it declares a CFS sentence) and 67% recall (i.e., 67% of true CFSs are classified as CFSs). The classification models had better accuracy on NFS and CFS than UFS. This is not surprising, since UFS is between the other two classes and thus the most ambiguous. More detailed results and analyses based on data collected by an earlier date can be found in [12].

Figure 4 shows the performance of these methods combined with the words, POS tags, and entity type features under various dataset sizes (4, 000, 8, 000, ..., 20, 000). Each set contained the 1, 032 screening sentences. We observe that the performance of SVM remained stable when dataset size was increased whereas the performance of NBC got better. This can be explained by how SVM and NBC work. SVM may have already discovered the decision boundary of the problem space with $4,000 - 8,000$ training instances. Hence, more training instances afterwards did not change the boundary much. On the other hand, NBC kept updating the conditional probabilities when more training data became available. Since SVM is the best performer among all the methods, we conducted all ensuing analyses using SVM trained over the smaller (8, 231) training data.

We used SVM to rank the sentences by the method in Section 3.1. We measured the accuracy of the top-k sentences by several commonly-used measures, including Precision-at-k (P@k), AvgP (Average Precision), nDCG (Normalized Discounted Cumulative Gain). Table 3 shows these measure values for various k values. In general, ClaimBuster achieved excellent performance in ranking. For instance, for top 100 sentences, its precision is 0.96. This indicates ClaimBuster has a strong agreement with high-quality human coders on the check-worthiness of sentences.

Table 2: Comparison of NBC, SVM and RFC coupled with various feature sets, in terms of Precision (p), Recall (r) and F-measure (f). wavg denotes weighted average of corresponding measure across three classes.

algorithm	features	p_NFS	p_UFS	p_CFS	p_wavg	r_NFS	r_UFS	r_CFS	r_wavg	f_NFS	f_UFS	f_CFS	f_wavg
RFC	W	0.755	0.125	0.638	0.692	0.965	0.004	0.235	0.745	0.848	0.008	0.343	0.685
NBC	W	0.788	0	0.816	0.747	0.983	0	0.385	0.791	0.875	0	0.522	0.744
SVM	W	0.871	0.426	0.723	0.811	0.925	0.227	0.667	0.826	0.897	0.296	0.694	0.816
RFC	W_P	0.772	0.358	0.701	0.731	0.968	0.011	0.312	0.764	0.859	0.02	0.43	0.713
NBC	W_P	0.799	0	0.805	0.753	0.979	0	0.44	0.8	0.88	0	0.569	0.758
SVM	W_P	0.873	0.43	0.724	0.813	0.925	0.24	0.671	0.827	0.898	0.307	0.696	0.818
RFC	W_P_ET	0.77	0.238	0.665	0.715	0.964	0.008	0.298	0.758	0.856	0.016	0.411	0.706
NBC	W_P_ET	0.803	0	0.791	0.752	0.976	0	0.455	0.801	0.881	0	0.577	0.76
SVM	W_P_ET	0.873	0.427	0.723	0.813	0.925	0.24	0.67	0.827	0.898	0.307	0.695	0.817

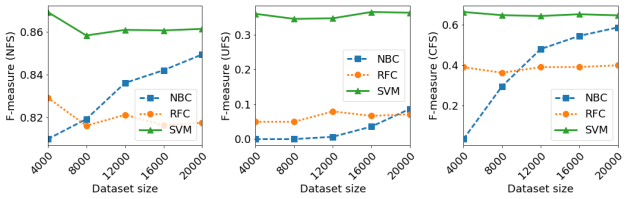

Figure 4: Change of performance with respect to data size

Table 3: Ranking accuracy: past presidential debates

k	P@k	AvgP	nDCG
10	1.000	1.000	1.000
25	1.000	1.000	1.000
50	0.980	0.995	0.987
100	0.943	0.979	0.956
200	0.916	0.955	0.931
300	0.848	0.937	0.874
400	0.764	0.915	0.801
500	0.679	0.897	0.827

3.5 Comparison with Subjectivity Classifiers

We also compared the performance of ClaimBuster with state-of-the-art subjectivity classifiers [20, 21]. Our hypothesis was that a subjectivity classifier can be used to separate NFS from UFS and CFS. However, experiment results showed that the subjectivity classifiers failed to filter out NFS. We used the OpinionFinder [16] package for classification. This tool provides two subjectivity classifiers [20, 21]. The first classifier [21] tags each sentence as either subjective or objective based on a model trained on the MPQA Corpus [17]. The second classifier [20] is a rule-based classifier. It optimizes precision at the expense of recall. That is, it classifies a sentence as subjective or objective only if it can do so with confidence. Otherwise, it labels the sentence as "unknown".

Table 4 shows the comparison between [21] and ClaimBuster. We used the 1032 screening sentences for this experiment. 574 NFS sentences were labeled as objective sentences and 44 CFS sentences were labeled as subjective sentences. This invalidates our hypothesis that a subjectivity classifier can be used to separate NFS sentences from UFS and CFS. Table 5 also shows similar comparison between ClaimBuster and [20].

Table 4: Predictions by Subjectivity Classifier [21]

	NFS	UFS	CFS
subjective	157	5	44
objective	574	58	194

Table 5: Predictions by Subjectivity Classifier [20]

	NFS	UFS	CFS
subjective	21	0	4
unknown	175	5	45
objective	535	58	189

4 CASE STUDY: 2016 U.S. PRESIDENTIAL ELECTION DEBATES

We compared ClaimBuster against the human fact-checkers at several fact-checking organizations. We are interested in testing the hypothesis that the claims picked by ClaimBuster are also more likely to be fact-checked by professionals. If the hypothesis is true, we can expect ClaimBuster to be effective in assisting professionals choose what to fact-check and thus improving their work efficiency.

4.1 Data Collection

There have been 12 Republican[18] and 9 Democratic primary debates in the 2016 U.S. presidential election. The debates featured as many as 11 Republican Party candidates and 5 Democratic Party candidates at the beginning, respectively. These debates took place between August, 2015 and April, 2016. We collected the transcripts of all these debates from several news media websites, including Washington Post, CNN, Times, and so on. There are a total of 30737 sentences in the 21 transcripts. We preprocessed these transcripts and identified the speaker of each sentence. Furthermore, we identified the role of the speaker. Sentences spoken by debate moderators were excluded from the study.

[16] http://mpqa.cs.pitt.edu/opinionfinder/
[17] http://mpqa.cs.pitt.edu/corpora/
[18] We only considered the "prime time" debates which included the more popular candidates.

4.2 Finding Check-worthy Factual Claims

We used ClaimBuster to calculate the check-worthiness scores of the sentences and thereby identify check-worthy factual claims. Figure 5 shows the distributions of ClaimBuster scores on all the sentences for both political parties. The distributions for the two parties are similar. One distinction is that the distribution for the Republican Party has a higher peak and a slightly thinner right tail than the distribution for the Democratic party. There are 776 check-worthy factual claims spoken by the Republicans with ClaimBuster scores over 0.5. This is 5.06% of all the sentences spoken by the Republican candidates. From Democrat candidates, there are 484(6.73%) sentences with ClaimBuster scores higher than 0.5.

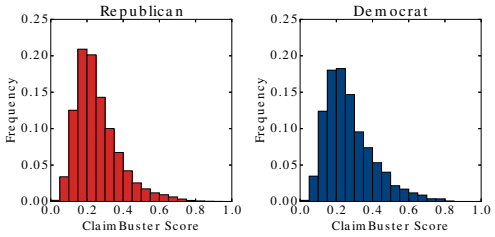

Figure 5: Distributions of ClaimBuster scores over all the sentences for both parties

Figure 6 shows the check-worthiness score distributions for the major candidates (nomination winners and runner-ups) from both parties. Among these four candidates, *Donald Trump* appears to have presented fewer highly check-worthy factual claims (ClaimBuster score ≥ 0.5) than the other three candidates. He has used more non-factual sentences (ClaimBuster score ≤ 0.3) compared to the other candidates.

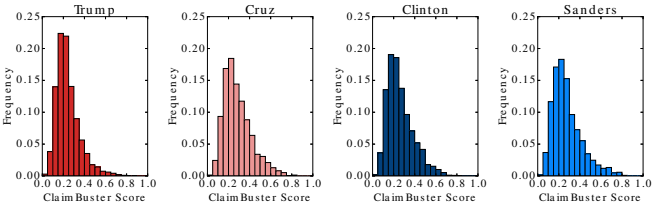

Figure 6: Distributions of ClaimBuster scores over all the sentences for the major candidates

4.3 Topic Detection

From each of the 21 debates, the 20 highest-scoring sentences were selected and manually placed in topic categories, a modified version of the most important problems (MIP) used by Gallup and other researchers for decades [17, 23, 28]. The major topics in the primary debates were: economy, crime, international affairs, immigration, health care, social issues, education, campaign finance, environment, Supreme Court, privacy and energy. Some of these topics were further broken down into subtopics. The 420 sample sentences were used to cultivate a list of keywords most often found for each of these topics. For example, the keywords for subtopic "abortion" were abortion, pregnancy and planned parenthood. Some topics had a small number of keywords, others had more than 20.

A topic-detection program is created to detect each debate sentence's topic. Provided a sentence, the program computes a score for each topic in our list based on presence of each topic's keywords in the sentence. The score is the total number of occurrences of such keywords. The sentence is assigned to the topic attaining the highest score among all the topics. However, if the highest score is lower than a threshold (two occurrences of topic keywords), the program does not assign any of the topics to the sentence. If there is a tie between two or more topics, the program uses the topic of the preceding sentence if it matches one of the tied topics. Otherwise, it randomly picks one of the tied topics.

In order to evaluate the above approach to detect topics, we created ground-truth data for one Republican debate and one Democratic debate. We only used sentences with at least 0.5 ClaimBuster score. In our ground-truth data for the Democratic debate, there are 52 sentences and 39 of them are labeled with a topic. The program detected topics for 27 of the 39 sentences and only one sentence was assigned with a incorrect topic. For the Republican debate ground-truth data, there are 62 sentences and 44 sentences are labeled with a topic. The program found topics for 30 out of the 44 sentences and 5 of these sentences were mis-classified.

We applied the topic detection program on all remaining sentences of these debates. The topics of the sentences allow us to gain better insight into the data. The results of our study which leverages the detected topics are reported in Section 4.5. The high accuracy of the topic-detection program on the ground-truth data gives us confidence on the results.

4.4 Verdict Collection

We used CNN and PolitiFact as the means for comparing ClaimBuster's results. These two organizations were selected because each identifies claims they judge to be worth checking and then rates each claim on a truthfulness scale. The verdicts for CNN are True, Mostly True, True but Misleading, False or It's Complicated. PolitiFact uses True, Mostly True, Half True, Mostly False, False and Pants on Fire (egregiously false). Other organizations focus on false or misleading claims only (Factcheck.org) or write about debate statements they found interesting or suspicious (Washington Post) which makes a comparison to ClaimBuster problematic.

For each of the 21 debates CNN and PolitiFact prepared a summary of the factual claims they chose to check and rendered a verdict on them. We collected all of these verdicts, 224 from CNN and 118 from PolitiFact.

Table 6 shows the scores given by ClaimBuster to the claims fact-checked by CNN and PolitiFact. The ClaimBuster average for sentences fact-checked by CNN is 0.433 compared to 0.258 for those sentences not selected by CNN, a statistically significant difference. Likewise, the ClaimBuster average for sentences checked by PolitiFact is 0.438 compared to 0.258 for those not selected, also a significant difference. The results of these comparisons demonstrate the utility of ClaimBuster in identifying sentences likely to contain important factual claims.

4.5 Results of Case Study

With the ClaimBuster score, topic and veracity of the sentences at hand, we study the relation among these and try to find answers to

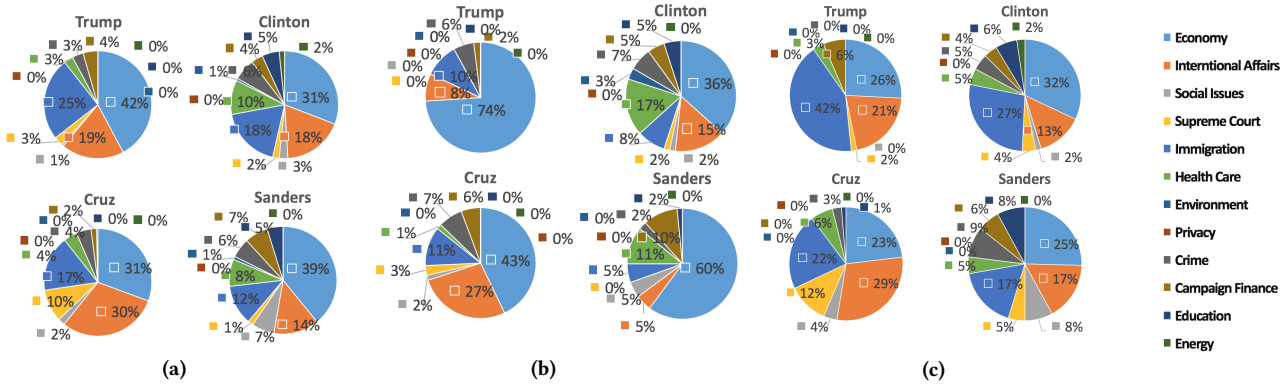

Figure 7: Distribution of topics over sentences from the major candidates. (a) all the sentences; (b) sentences scored low (≤ 0.3) by ClaimBuster; (c) sentences scored high (≥ 0.5) by ClaimBuster

Table 6: ClaimBuster score differences between sentences fact-checked and those not chosen for checking

Platforms	avg(YES)	avg(NO)	t-value	p-value
CNN	0.433	0.258	21.137	1.815E-098
PolitiFact	0.438	0.258	16.362	6.303E-060

questions such as which candidate presented more factual claims pertaining to a certain topic compared to others and so on.

Figure 7(a) shows the distribution of topics among sentences by each major candidate in the race. *Bernie Sanders* was the most vocal on *Social Issues* among the candidates. *Ted Cruz* spoke significantly more on *International Affairs* compared to other candidates.

We analyzed the check-worthiness of the sentences of each topic. Figure 7(b) shows the topic distribution of sentences having ClaimBuster score ≥ 0.5. This figure explains how often the candidates used factual claims while speaking about different topics. For example, both *Donald Trump* and *Bernie Sanders* presented significantly more check-worthy factual claims relating to the *Economy* compared to their debate competitors.

Figure 7(c) shows the topic distribution of sentences having ClaimBuster score ≤ 0.3. This figure explains how much the candidates spoke about different topics without presenting factual claims. One observation derived from Figures 7(b) and 7(c) is that Republican candidates spoke about *Health Care* but used fewer factual claims regarding this topic. On the other hand, Democratic candidate *Hillary Clinton* presented factual statements related to *Environment* rather than presenting non-factual statements.

Figure 8 shows the topic distributions of CNN, PolitiFact sentences as well as of highly check-worthy factual sentences (ClaimBuster score ≥ 0.5). This figure signifies that there are strong similarities between ClaimBuster and the fact-checking organizations. ClaimBuster tends to give high scores to the topics which CNN and PolitiFact tend to choose for fact checking. For example, all three have about 50 percent of the fact checks (or high ClaimBuster scores) associated with *Economy*, about 14 percent for *International Affairs*, about 10 percent for *Immigration* and 4 percent for *Crime*. One topic where ClaimBuster showed a difference with the human fact-checkers was *Social Issues*. That topic represented about 9 percent of the CNN and PolitiFact fact-checks but only about 2 percent of the highly scored ClaimBuster sentences.

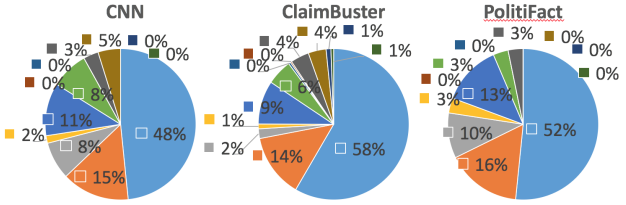

Figure 8: Comparing topic distributions of sentences checked by CNN, PolitiFact and sentences scored high (≥ 0.5) by ClaimBuster

5 CURRENT STATUS OF CLAIMBUSTER

Sections 3 and 4 present the claim spotter component of ClaimBuster. This section introduces the current status of other components in the system. The system is hosted at http://idir.uta.edu/claimbuster and its features are being gradually expanded. Figure 9 depicts the system architecture of ClaimBuster. It consists of several integral components, as follows.

Claim Monitor: This component continuously monitors and retrieves texts from a variety of sources, upon which claim spotting is applied to discover important factual claims. At present, the system monitors the following sources.

Broadcast Media: ClaimBuster uses a decoding device to extract closed captions in broadcasted TV programs. This was used for our live coverage of all twenty-one primary election debates and four general election debates of the 2016 U.S. presidential election. Figure 10 shows the coverage of one of the debates. Sentences in the transcript are highlighted in different shades of blue proportional to their check-worthiness scores. The platform allows a user to order the sentences by time or by score and to use a slider to specify the minimum score for sentences to be highlighted. It also provides interactive visualizations of the scores of the sentences (omitted in the figure).

Social Media: ClaimBuster has been continuously monitoring a list of 2220 Twitter accounts (U.S. politicians, news and media

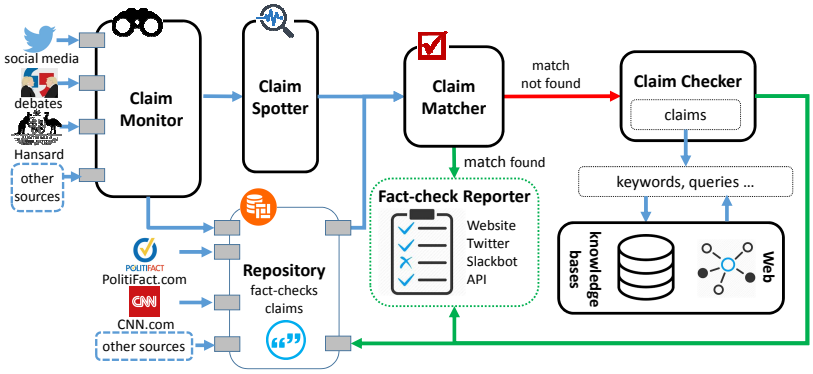

Figure 9: Major components of ClaimBuster (under continuous improvement and development)

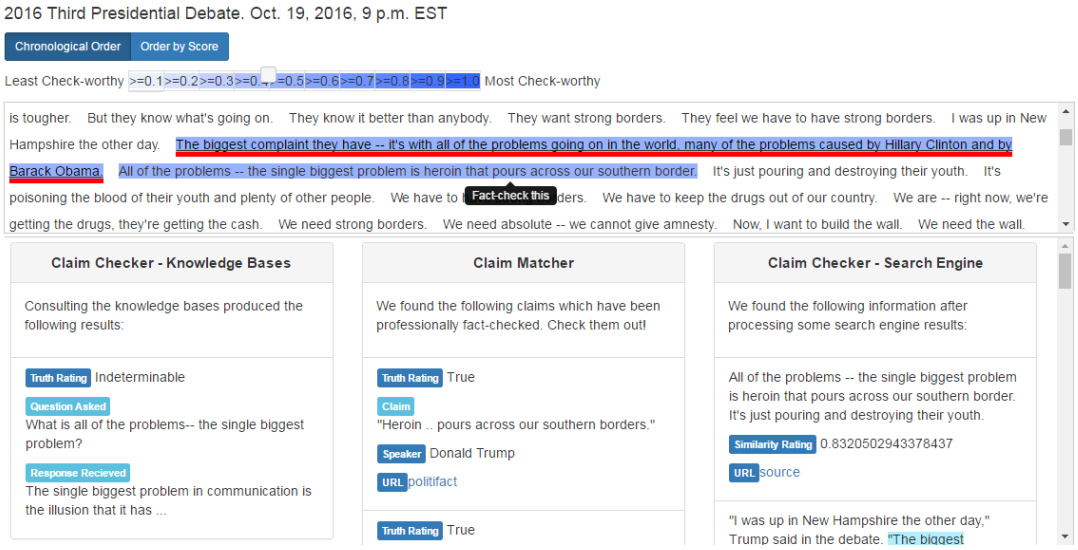

Figure 10: ClaimBuster GUI on a presidential debate

organizations) using the Twitter streaming API. We evaluated the claim spotting model on a randomly selected set of 1,000 tweets (after filtering out non politics-related tweets and junk tweets [1]) among 27 million collected tweets. These tweets were manually labelled as either CFS (39 of them) or not. We applied the claim spotting model on the tweets and ranked the tweets using their scores. The accuracy of the ranking, measured in Precision-at-k (P@k), Average Precision (AvgP), and Normalized Discounted Cumulative Gain (nDCG), can be found in Table 7. Although the claim spotting model was trained using a labeled dataset of presidential debates, we find that the model achieved weaker but still satisfactorily accurate results on similar politics-related text outside of presidential debates such as these tweets.

Websites: ClaimBuster also gathers data from websites. For instance, it continuously monitors "Hansard" – the transcripts of proceedings of the Australian parliament.

Claim Matcher: Given an important factual claim identified by the claim spotter, the *claim matcher* searches a fact-check repository and returns those fact-checks matching the claim. The repository was curated from various fact-checking websites. The system has two approaches to measuring the similarity between a claim and a fact-check. One is based on the similarity of tokens and the other is based on semantic similarity.

Table 7: Ranking accuracy of ClaimBuster on Twitter data

k	P@k	AvgP	nDCG
10	0.500	0.576	0.855
25	0.480	0.543	0.832
50	0.300	0.458	0.821
100	0.190	0.352	0.806
200	0.145	0.258	0.768
500	0.070	0.161	0.749
1000	0.039	0.106	0.735

Claim Checker: Given a claim, the *claim checker* collects supporting or debunking evidence from knowledge bases (e.g., Wolfram Alpha [19]) and the Web (e.g., Google answer boxes). If any clear discrepancies between the returned answers and the claim exist, a verdict may be derived and presented to the user. Meanwhile, the factual claim itself is sent to Google as a general search query. The claim checker then parses the search result and downloads the web page for each top result. Within each such page, it finds

[19] http://products.wolframalpha.com/api/

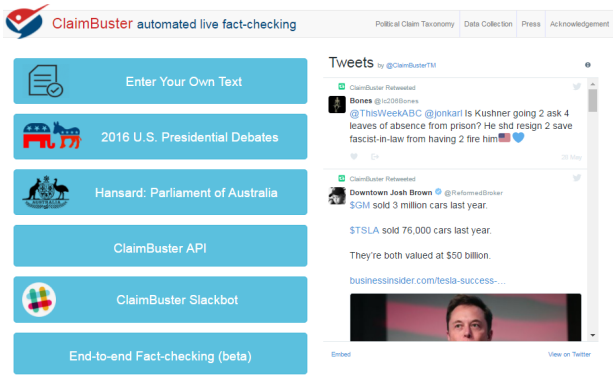

Figure 11: Homepage of ClaimBuster website

sentences matching the claim. The matching sentences and a few of their surrounding sentences are then grouped together into a *context*. The contexts, answers returned from Wolfram Alpha and Google answer boxes, as well as any verdicts derived from those answers form the supporting or debunking evidence for the claim. The evidence is reported to the user, as follows.

Fact-check Reporter: The *fact-check reporter* synthesizes a report (e.g., the bottom part of Figure 10) by combining the aforementioned evidence and delivers it to users through the project website. Furthermore, ClaimBuster also delivers the claim spotter scores on claims through a variety of channels, including its website, Twitter account, API, and Slackbot. Its Twitter account (@ClaimBusterTM) retweets the highly-scored tweets from politicians and organizations and posts highly-scored claims from live events such as the presidential debates. To this date, @ClaimBusterTM has retweeted and posted about 13K check-worthy factual claims. A Slackbot has been developed for users to supply their own text (i.e., directly as input or through a shared Dropbox folder) and receive the claim spotter score and fact-check report for that piece of text. The Slackbot has been published in the public Slack App directory and can also be installed by clicking the "ClaimBuster Slackbot" button on the project website (Figure 11). We also made available a public ClaimBuster API (note the button in Figure 11) to allow developers create their own fact-checking applications.

6 CONCLUSION

ClaimBuster can quickly extract and order sentences in ways that will aid in the identification of important factual claims. We used the 2016 U.S. presidential election debates to compare the results of our automated factual claim tool against the judgments of professional journalism organizations. Overall, we found that sentences selected by both CNN and PolitiFact for fact checking had ClaimBuster scores that were significantly higher (were more check-worthy) than sentences not selected for checking. We are also using ClaimBuster to check content on popular social platforms where much political information is being generated and shared. But there is still much work to be done. Discrepancies between the human checkers and the machine have provided us with avenues for improvement of the algorithm. A next step is the adjudication of identified check-worthy claims. A repository of already-checked facts would be a good starting point. Each of these areas are demanding and worthy of attention by the growing field of computational journalism.

Acknowledgements: The ClaimBuster project is partially supported by NSF grants IIS-1408928, IIS-1565699 and a Knight Prototype Fund from the Knight Foundation. Any opinions, findings, and conclusions or recommendations expressed in this publication are those of the authors and do not necessarily reflect the views of the funding agencies.

REFERENCES

[1] Fatma Arslan. 2015. *Detecting Real-time Check-worthy Factual Claims in Tweets Related to U.S. Politics.* Master's thesis. University of Texas at Arlington.
[2] Prakhar Biyani, Sumit Bhatia, Cornelia Caragea, and Prasenjit Mitra. 2014. Using non-lexical features for identifying factual and opinionative threads in online forums. *Knowledge-Based Systems* 69 (2014), 170–178.
[3] Carlos Castillo, Marcelo Mendoza, and Barbara Poblete. 2011. Information Credibility on Twitter. In *WWW*. 675–684.
[4] Giovanni Luca Ciampaglia, Prashant Shiralkar, Luis M. Rocha, Johan Bollen, Filippo Menczer, and Alessandro Flammini. 2015. Computational Fact Checking from Knowledge Networks. *PLOS ONE* 10, 6 (June 2015), 1–13.
[5] Hugo De Burgh. 2008. *Investigative journalism.* Routledge.
[6] Yuxiao Dong, Jing Zhang, Jie Tang, Nitesh V. Chawla, and Bai Wang. 2015. CoupledLP: Link Prediction in Coupled Networks. In *KDD*. 199–208.
[7] Adrien Friggeri, Lada Adamic, Dean Eckles, and Justin Cheng. 2014. Rumor Cascades. In *ICWSM*.
[8] FullFact.org. 2016. The State of Automated Factchecking. https://fullfact.org/blog/2016/aug/automated-factchecking/. (2016).
[9] Jonathan Gray, Lucy Chambers, and Liliana Bounegru (Eds.). 2012. *The Data Journalism Handbook.* Oreilly & Associates Inc. http://datajournalismhandbook.org/
[10] Naeemul Hassan et al. 2017. ClaimBuster: The First-ever Automated, Live Fact-checking System. In *VLDB*.
[11] Naeemul Hassan, Bill Adair, James T. Hamilton, Chengkai Li, Mark Tremayne, Jun Yang, and Cong Yu. 2015. The Quest to Automate Fact-Checking. In *Computation+Journalism Symposium*.
[12] Naeemul Hassan, Chengkai Li, and Mark Tremayne. 2015. Detecting Check-worthy Factual Claims in Presidential Debates. In *CIKM*. 1835–1838.
[13] Naeemul Hassan, Mark Tremayne, Fatma Arslan, and Chengkai Li. 2016. Comparing Automated Factual Claim Detection Against Judgments of Journalism Organizations. In *Computation+Journalism Symposium*.
[14] Jure Leskovec, Lars Backstrom, and Jon Kleinberg. 2009. Meme-tracking and the Dynamics of the News Cycle. In *KDD*.
[15] Yaliang Li, Jing Gao, Chuishi Meng, Qi Li, Lu Su, Bo Zhao, Wei Fan, and Jiawei Han. A Survey on Truth Discovery. *SIGKDD Explor. Newsl.* 17, 2 (????), 1–16.
[16] Ryan N. Lichtenwalter and Nitesh V. Chawla. 2012. Vertex Collocation Profiles: Subgraph Counting for Link Analysis and Prediction. In *WWW*. 1019–1028.
[17] Maxwell E McCombs and Donald L Shaw. 1972. The agenda-setting function of mass media. *Public opinion quarterly* 36, 2 (1972), 176–187.
[18] John Platt et al. 1999. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers* 10, 3 (1999).
[19] Vahed Qazvinian, Emily Rosengren, Dragomir R. Radev, and Qiaozhu Mei. 2011. Rumor Has It: Identifying Misinformation in Microblogs. In *EMNLP*. 1589–1599.
[20] Ellen Riloff and Janyce Wiebe. 2003. Learning extraction patterns for subjective expressions. In *EMNLP*. 105–112.
[21] Ellen Riloff, Janyce Wiebe, and William Phillips. 2005. Exploiting subjectivity classification to improve information extraction. In *AAAI*. 1106–1111.
[22] Baoxu Shi and Tim Weninger. 2016. Discriminative Predicate Path Mining for Fact Checking in Knowledge Graphs. *Knowledge-Based Systems* 104, C (July 2016), 123–133.
[23] Tom W Smith. 1980. America's most important problem-a trend analysis, 1946–1976. *Public Opinion Quarterly* 44, 2 (1980), 164–180.
[24] Andreas Vlachos and Sebastian Riedel. 2014. Fact Checking: Task definition and dataset construction. In *ACL*. 18–22.
[25] Janyce Wiebe and Ellen Riloff. 2005. Creating subjective and objective sentence classifiers from unannotated texts. In *CICLing*. 486–497.
[26] Tamar Wilner. 2014. Meet the robots that factcheck. *Columbia Journalism Review* (September-October 2014).
[27] Hong Yu and Vasileios Hatzivassiloglou. 2003. Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences. In *EMNLP*. 129–136.
[28] Jian-Hua Zhu. 1992. Issue competition and attention distraction: A zero-sum theory of agenda-setting. *Journalism & Mass Communication Quarterly* 69, 4 (1992), 825–836.