The Quest to Automate Fact-Checking

Naeemul Hassan [1] Bill Adair [2] James T. Hamilton [3] Chengkai Li [1]
Mark Tremayne [1] Jun Yang [2] Cong Yu [4]

[1]University of Texas at Arlington, [2]Duke University, [3]Stanford University, [4]Google Research

1. INTRODUCTION

The growing movement of political fact-checking plays an important role in increasing democratic accountability and improving political discourse [7, 3]. Politicians and media figures make claims about "facts" all the time, but the new army of fact-checkers can often expose claims that are false, exaggerated or half-truths. The number of active fact-checking websites has grown from 44 a year ago to 64 this year, according the Duke Reporters's Lab. [1]

The challenge is that the human fact-checkers frequently have difficulty keeping up with the rapid spread of misinformation. Technology, social media and new forms of journalism have made it easier than ever to disseminate falsehoods and half-truths faster than the fact-checkers can expose them.

There are several reasons that the falsehoods frequently outpace the truth. One reason is that fact-checking is an intellectually demanding and laborious process. It requires more research and a more advanced style of writing than ordinary journalism. The difficulty of fact-checking, exacerbated by a lack of resources for investigative journalism, leaves many harmful claims unchecked, particularly at the local level.

Another reason is that fact-checking is time-consuming. It takes about one day to research and write a typical article, which means a lot of time can lapse after the political message. Even if the fact-check has already been published, the voter must undertake research to look it up. This "gap" in time and availability limits the effectiveness of fact-checking.

Computation may hold the key to far more effective and efficient fact-checking, as Cohen et al. [1, 2] and Diakopolous [2] have pointed out. Over and over again, computing has reshaped journalism. Tasks that required huge amounts of manual labor such as analyzing data and finding patterns and relationships are now accomplished with ease. There is little doubt that computers can substantially aid fact-checking too.

The eternal quest, the "Holy Grail", is a completely automatic fact-checking platform that can detect a claim as it appears in real time, and instantly provide the voter with a rating about its accuracy. It makes its calls by consulting databases of already checked claims, and by conducting novel analysis of relevant data from reputable sources.

In this paper, we advocate the pursuit of the "Holy Grail" and make a call to arms to the computing and journalism communities. We discuss the technical challenges we will face in automating fact-checking and potential solutions. The "Holy Grail" may remain far beyond our reach for many, many years to come.

But in pursuing this ambitious goal, we can help fact-checking and improve the political discourse. One such advancement is our own progress on ClaimBuster, a tool that helps journalists find political claims to fact-check. We will use it on the presidential debates of U.S. Election 2016. We envision, during a debate, for every sentence spoken by the candidates and extracted into transcripts, ClaimBuster immediately determines if the sentence has a factual claim and whether its truthfulness is important to the public.

2. LIMITATIONS OF CURRENT PRACTICES OF FACT-CHECKING

Fact-checking is difficult and time-consuming for journalists, which creates a significant gap between the moment a politician makes a statement and when the fact-check is ultimately published.

The growth of fact-checking has been hampered by the nature of the work. It is time-consuming to find claims to check. Journalists have to spend hours going through transcripts of speeches, debates and interviews to identify claims they will research.

Also, fact-checking requires advanced research techniques. While ordinary journalism can rely on simple "on-the-one-hand, on-the-other-hand" quotations, a fact-check requires more thorough research so the journalist can determine the accuracy of a claim.

Fact-checking also requires advanced writing skills that go beyond "just the facts" to persuade the reader whether the statement was true, false or somewhere in between. Fact-checking is a new form that has been called "reported conclusion" journalism.

Those factors mean that fact-checking often takes longer to produce than traditional journalism, which puts a strain on staffing and reduces the number of claims that can be

[1] http://reporterslab.org/snapshot-of-fact-checking-around-the-world-july-2015/

[2] http://towknight.org/research/thinking/scaling-fact-checking/

checked. It also creates a time gap between the moment the statement was made and when the fact-check is ultimately published. That can take as little as 15 to 30 minutes for the most simple fact-check to a full day for a more typical one. A complicated fact-check can take two or more days. (By contrast, Leskovec, Backstrom and Kleinberg [6] found a meme typically moves from the news media to blogs in just 2.5 hours.)

For voters, that means a long gap between the politician's claim and a determination whether it was true. The voters don't get the information when they really need it. They must wait and look up on a fact-checking site to find out if the claim was accurate. This is one of several factors that emboldens politicians to keep repeating claims even when they are false.

Another limitation is the outdated nature of the fact-checkers' publishing platforms. Many fact-checking sites still use older content management systems built for newspapers and blogs that are are not designed in a modern style for structured journalism. This limits how well they can be used in computational projects.

3. THE "HOLY GRAIL"

We should not be surprised if we can get very close but never reach the "Holy Grail". A fully automated fact-checker calls for fundamental breakthroughs in multiple fronts and, eventually, it represents a form of Artificial Intelligence (AI). As remote and intangible as AI may have appeared initially, though, in merely 60 years scientists have made leaps and bounds that profoundly changed our world forever. The quest for the "Holy Grail" of fact-checking will likewise drive us to constantly improve this important journalistic activity.

The Turing test [9] was proposed by Alan Turing as a way of gauging a machine's ability to exhibit artificial intelligence. Although heavily criticized, the concept has served well in helping advance the field. Similarly, we need explicit and tangible measures for assessing the ultimate success of a fact-checking machine. The "Holy Grail" is a computer-based fact-checking system bearing the following characteristics:

Fully automated: It checks facts without human intervention. It takes as input the video/audio signals and texts of a political discourse and returns factual claims and a truthness rating for each claim (e.g., the Truth-O-Meter by PolitiFact).

Instant: It immediately reaches conclusions and returns results after claims are made, without noticeable delays.

Accurate: It is equally or more accurate than any human fact-checker.

Accountable: It self-documents its data sources and analysis, and makes the process of each fact-check transparent. This process can then be independently verified, critiqued, improved, and even extended to other situations.

Such a system mandates many complex steps–extracting natural language sentences from textual/audio sources; separating factual claims from opinions, beliefs, hyperboles, questions, and so on; detecting topics of factual claims and discerning which are the "check-worthy" claims; assessing the veracity of such claims, which itself requires collecting information and data, analyzing claims, matching claims with evidence, and presenting conclusions and explanations. Each step is full of challenges. We now discuss in more detail these challenges and potential solutions.

3.1 Computational Challenges

On the computational side, there are mainly two fundamental challenges. One is to understand what one says. Computer scientists have made leaps and bounds in speech recognition and Natural Language Processing (NLP). But these technologies are far from perfect. The other challenge lies in our capability of collecting sufficient evidence for checking facts. We are in the big-data era. A huge amount of useful data is accessible to us and more is being made available at every second. Semantic web, knowledge base, database and data mining technologies help us link together such data, reason about the data, efficiently process the data and discover patterns. But, what is being recorded is still tiny compared to the vast amount of information the universe holds. Below we list some of the more important computational hurdles to solve.

Finding claims to check

—Going from raw audio/video signals to natural language. Extracting contextual information such as speaker, time, and occasion.

—Defining "checkable" and "check-worthy" of claims. Is the claim factual (falsifiable) or is it opinion? Should or can we check opinions? How "interesting" is the claim? How do we balance "what the public should know" and "what the public wants to consume"? Can these judgements be made computationally?

—Extracting claims from natural language. What to do when a claim spans multiple sentences? What are the relevant features useful for determining whether a claim is "checkable" or "check-worthy"?

Getting data to check claims

—We should consider at least two types of data sources: 1) claims already checked by various organizations; 2) unstructured, semi-structured and structured data sources that provide raw data useful for checking claims, e.g., voting records, government budget, historical stock data, crime records, weather records, sports statistics, and Wikipedia.

—Evaluating quality and completeness of sources.

—Matching claims with data sources. This requires structure/metadata in the database of already checked claims, as well as data sources.

—Synthesizing and corroborating multiple sources.

—Cleansing data. Given a goal (e.g., to verify a particular claim), help journalists decide which data sources–or even which data items in a database–are worthy investigating as high priority.

Checking claims

—How to remove (sometimes intentional) vagueness, how to spot cherry-picking of data (beyond correctness), how to evaluate and how to come up with convincing counterarguments using data [12, 11, 10].

—The methods in [12, 11, 10] rely on being able to cast a claim as a mathematical function that can be evaluated over structured data. Who translate a claim into this function? Can the translation process be automated?

—Fact verification may need human participation (e.g., social media as social sensors) or even crowdsourcing (e.g., checking whether a bridge really just collapsed). Can a computer system help coordinate and plan human participation on an unprecedented level? How to remove bias and do quality control of human inputs? Should such a system be even considered fully automated?

Monitoring and anticipating claims

—Given evolving data, we can monitor when a claim turns

false/true [5, 10]. Can we anticipate what claims may be made soon? That way, we can plan ahead and be proactive.
—Challenges in scalable monitoring and parallel detection of a massive number of claim types/templates.

3.2 Journalistic Challenges

A major barrier to automation is the lack of structured journalism in fact-checking. Although there's been tremendous growth in the past few years – 20 new sites around the world just in the last year, according to the Duke Reporters' Lab – the vast majority of the world's fact-checkers are still relying on old-style blog platforms to publish their articles. That limits the articles to a traditional headline and text rather than a newer structured journalism approach that would include fields such as statement, speaker and location that would allow for real-time matching. There are no standards for data fields or formatting. The articles are just published as plain text.

There also is no single repository where fact-checks from various news organizations are catalogued. They are kept in the individual archives of many different publications, another factor that makes real-time matching difficult. Another journalistic barrier is the inconsistency of transparency. Some fact-checkers distill their work to very short summaries, while others publish lengthy articles with many quotations and citations.[3] The lack of structure, the absence of a repository and the inconsistency in publishing provides a lack of uniformity for search engines, which do not distinguish fact-checks from other types of editorial content in their search results.

Another challenge is the length of time it takes to publish more difficult fact-checks and to check multiple claims from the same event. PolitiFact, for example, boasted that it published 20 separate checks from the Aug. 6 Republican presidential debate. But it took six days for it to complete all of those checks. [4]

4. CLAIMBUSTER

ClaimBuster is a tool that helps journalists find claims to fact-check. Figure 1 is the screenshot of the current version of ClaimBuster. For every sentence spoken by the participants of a presidential debate, ClaimBuster determines whether the sentence has a factual claim and whether its truthfulness is important to the public. As shown in Figure 1, to the left of each sentence there is a score ranging from 0 (least likely an important factual claim) to 1 (most likely). The calculation is based on machine learning models built from thousands of sentences from past debates labeled by humans. The ranking scores help journalists prioritize their efforts in assessing the varacity of claims. ClaimBuster will free journalists from the time-consuming task of finding check-worthy claims, leaving them with more time for reporting and writing. Ultimately, ClaimBuster can be expanded to other discourses (such as interviews and speeches) and also adapted for use with social media.

4.1 Classification and Ranking

We model ClaimBuster as a classifier and ranker and we take a supervised learning approach to construct it. We categorize sentences in 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 some 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 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 a Support Vector Machine (SVM). We treat CFSs as positive examples and both NFSs and UFSs as negative examples. SVM finds a decision boundary between the two types of training examples. Following Platt's scaling technique [8], for a given sentence x to be classified, we calculate the posterior probability $P(class = CFS|x)$ using the SVM's decision function. The probability scores of all sentences are used to rank them.

4.2 Data Collection

We constructed a labeled dataset of sentences spoken by presidential candidates in all past general election presidential debates. Each sentence is given one of three possible labels– NFS, UFS, CFS.

[3] http://reporterslab.org/study-explores-new-questions-about-quality-of-global-fact-checking/
[4] http://www.politifact.com/truth-o-meter/article/2015/aug/12/20-fact-checks-republican-debate/

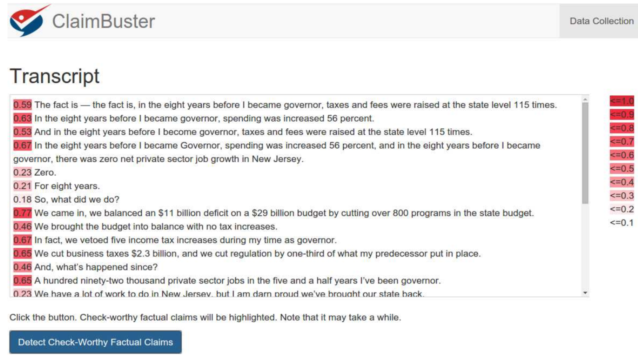

Figure 1: ClaimBuster

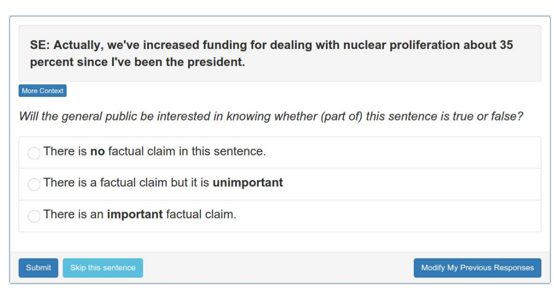

Figure 2: Data Collection Interface

Table 1: Performance

	Precision	Recall	F-measure
NFS	0.90	0.96	0.93
UFS	0.65	0.26	0.37
CFS	0.79	0.74	0.77

Table 2: Ranking Accuracy: Past Presidential Debates

k	P@k	AvgP	nDCG
10	1.000	0.024	1.000
25	1.000	0.059	1.000
50	1.000	0.118	1.000
100	0.960	0.223	0.970
200	0.940	0.429	0.951
300	0.853	0.575	0.881
400	0.760	0.667	0.802
500	0.690	0.737	0.840

There have been a total of 30 presidential debates in the past. We parsed the debate transcripts and extracted 23075 sentences spoken by the candidates. Furthermore, we only kept the 20788 sentences that have at least 5 words.

To label the sentences, we developed a data collection website. Journalists, professors and university students were invited to participate. A participant was given one sentence at a time and was asked to label it with one of the three possible options as shown in Figure 2, corresponding to the three labels (NFS, UFS, CFS).

In 3 months, we accumulated 226 participants. To detect spammers and low-quality participants, we used 600 screening sentences, picked from all debate episodes. Three experts agreed upon their labels. 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 selected from the pool. The participants were ranked by the degree of agreement on screening sentences between them and the three experts. The top 30% participants were considered top-quality participants. There was a reward system to encourage high quality participants. For training and evaluating our classification models, we only used a sentence if its label was agreed upon by two top-quality participants. Thereby we got 8015 sentences (5860 NFSs, 482 UFSs, 1673 CFSs).

4.3 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. 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 rare words that appear in less than three sentences, we got 6130 words. We did not apply stemming or stopword removal.

Part-of-Speech (POS) Tag: We applied NLTK POS tagger on all 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).

Table 3: Ranking Accuracy: 2015 Republican Debate

k	P@k	AvgP	nDCG
10	0.400	0.046	0.441
20	0.450	0.084	0.456
30	0.367	0.098	0.401
40	0.325	0.111	0.368
50	0.300	0.122	0.346
60	0.300	0.139	0.356
70	0.300	0.154	0.390
80	0.275	0.159	0.401
90	0.267	0.169	0.422
100	0.270	0.184	0.452

Entity Type: We used AlchemyAPI to extract entities from sentences. There are 2727 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 6201 features in total. To avoid over-fitting and attain a simpler model, 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. We observed that unsurprisingly POS tag CD (Cardinal Number) is the best feature–check-worthy factual claims are more likely to contain numeric values (45% of CFS sentences in our dataset contain numeric values) and non-factual sentences are less likely to contain numeric values (6% of NFS sentences in our dataset contain numeric values).

4.4 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. SVM had the best accuracy in general. We experimented with various combinations of the extracted features. Table 1 shows the performance of SVM using words and POS tag features. On the CFS class, ClaimBuster achieved 79% precision (i.e., it is accurate 79% of the time when it declares a CFS sentence) and 74% recall (i.e., 74% 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 [4].

We used SVM to rank all 8015 sentences (cf. Section 4.2) by the method in Section 4.1. 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 2 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.

5. CASE STUDY: 2015 GOP DEBATE

The first Republican primary debate of 2015 (the top-ten polling candidates) provided an opportunity for a near real-time test of ClaimBuster. Closed captions of the debate on Fox News were converted to a text file via TextGrabber, a device for the hearing impaired, and run through ClaimBuster. It parsed 1,393 sentences spoken by the candidates and moderators. ClaimBuster's scores on these sentences ranged from a low of 0.045 to a high of 0.861 with a mean of 0.263. Most sentences (87%) scored below 0.40.

We can compare ClaimBuster's identification of check-worthy factual claims against the judgement of professional journalists and fact checkers. Note that the accuracy of ClaimBuster is affected by the quality of TextGrabber in extracting closed captions. In general, the extracted closed captions demonstrated satisfactory quality. We also performed the same experiments using a human-refined version of the debate transcript and observed slightly better accuracy from ClaimBuster. Due to space limitations, we omit discussing that result.

Table 4 shows scores ClaimBuster gave to the claims fact-checked by CNN.[5] The average for these 6 was 0.457 compared to 0.262 for those sentences not selected by CNN, a significant difference (t=3.83, p<.001). As the transcript is from closed captions, some words and sentences are misspelled and missing (e.g., Claim 6 not found in the TextGrabber transcript). Note that Claim 4 spans over two sentences.

There were 9 sentences in our data that were selected for checking by FactCheck.org.[6] Due to space limitation, we do not show the text of the claims. These sentences averaged 0.558 compared to 0.261 for those not checked, a significant difference (t=7.23, p<.00001). PolitiFact[7] has checked 20 facts. The average ClaimBuster score for those sentences is 0.433 compared to 0.260 for those not checked by PolitiFact, also significant (t=6.67, p<.00001).

In addition to the claims fact-checked by CNN, FactCheck.org and PolitiFact we also had a larger "buffet" file from PolitiFact.[8] This file contained 59 claims from the debate which PolitiFact employees marked as possible items for fact-checking. We used ClaimBuster to rank these claims with respect to all the sentences (1,393) in the transcript. Table 3 shows the quality of this ranking in terms of P@K, AvgP and nDCG, in the same way we used these measures to evaluate ClaimBuster's ranking accuracy on past debate sentences.

Overall, sentences receiving a high ClaimBuster score were much more likely to have been checked by professionals than those with low scores. Most of those checked by CNN, FactCheck.org and PolitiFact (27 of 38 or 71%) appeared in the top 250 of 1,393 sentences. A lower percentage of sentences associated with items in the PolitiFact "buffet" file (53 of 83 or 64%) appeared in ClaimBuster's top 250. This is not surprising since these items were merely placed on the buffet by individual employees and not necessarily selected by the group for checking.

There were still many sentences ranked high by ClaimBuster and not chosen for fact-checking by these organizations. Reasons may include 1) the claims were previously made and checked; 2) they are not considered factual or important by the checker; 3) time and resource limitations.

[5]http://www.cnn.com/2015/08/06/politics/republican-debate-fact-check/

[6]http://www.factcheck.org/2015/08/factchecking-the-gop-debate-late-edition/

[7]http://www.politifact.com/truth-o-meter/article/2015/aug/12/20-fact-checks-republican-debate/

[8]PolitiFact, List of possible claims to check, Republican presidential debate, Aug. 6, 2015.

Table 4: ClaimBuster Performance on CNN-checked claims

Claim	Associated sentence(s)[From TextGrabber]	Score
1	Part of this iranian deal was lifting the international sanctions on general sulemani.	0.415
2	I would go on to add – >> you don't favor – >> i have never said that.	0.511
3	A majority of the candidates on this stage supported amnesty.	0.295
4	Timely the medicaid is growing at one of the lowest rates in the country.	0.534
4	We went from $8 billion in the hole to $5 million in the black.	0.773
5	And the mexican government is much smarter, much sharper, much more cunning and they send the bad ones over because they don't want to pay for them.	0.215
6	[Not found in the transcript]	N/A

6. CONCLUSION

Live, fully-automated fact checking may remain an unattainable ideal but serves as a useful guidepost for researchers in computational journalism. Already progress on the first steps of fact checking has been achieved. Our ClaimBuster tool, still imperfect, can quickly extract and order sentences in ways that will aid in the identification of important factual claims. Sentences from the recent GOP debate transcript with high ClaimBuster scores were more likely to be identified by experts as ones needing checking than those with lower scores. 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 in time for upcoming 2016 debates. An even bigger step will be the adjudication of identified check-worthy claims. A repository of already-checked facts would be good starting point. We are also interested in using ClaimBuster to check content on popular social platforms where much political information is being generated and shared. Each of these areas are demanding and worthy of attention by the growing field of computational journalism.

7. REFERENCES

[1] S. Cohen, J. T. Hamilton, and F. Turner. Computational journalism. *CACM*, 54(10):66–71, Oct. 2011.

[2] S. Cohen, C. Li, J. Yang, and C. Yu. Computational journalism: A call to arms to database researchers. In *CIDR*, 2011.

[3] L. Graves. *Deciding What's True: Fact-Checking Journalism and the New Ecology of News.* PhD thesis, COLUMBIA UNIVERSITY, 2013.

[4] N. Hassan, C. Li, and M. Tremayne. Detecting check-worthy factual claims in presidential debates. In *CIKM*, 2015.

[5] N. Hassan, A. Sultana, Y. Wu, G. Zhang, C. Li, J. Yang, and C. Yu. Data in, fact out: Automated monitoring of facts by FactWatcher. *PVLDB*, 7(13):1557–1560, 2014.

[6] J. Leskovec, L. Backstrom, and J. Kleinberg. Meme-tracking and the dynamics of the news cycle. In *KDD*, 2009.

[7] B. Nyhan and J. Reifler. The effect of fact-checking on elites: A field experiment on us state legislators. *American Journal of Political Science*, 59(3):628–640, 2015.

[8] J. Platt et al. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3), 1999.

[9] A. M. Turing. Computing machinery and intelligence. *Mind*, 59(236):433–460, 1950.

[10] B. Walenz et al. Finding, monitoring, and checking claims computationally based on structured data. In *Computation+Journalism Symposium*, 2014.

[11] Y. Wu, P. K. Agarwal, C. Li, J. Yang, and C. Yu. Toward computational fact-checking. In *PVLDB*, 2014.

[12] Y. Wu, B. Walenz, P. Li, A. Shim, E. Sonmez, P. K. Agarwal, C. Li, J. Yang, and C. Yu. iCheck: computationally combating "lies, d-ned lies, and statistics". In *SIGMOD*, 2014.