A Functional and Scale-able User Platform for Automatic Fake News Detection

Sari Stissi
Department of Computer Science
Earlham College
Richmond, Indiana, United States
sestissi16@earlham.edu

ABSTRACT

No one is above the harm that can come from misinformation, and possible solutions to the issue are hindered because not everyone can access the internet equally. That's why it's important to fight against the spread of misinformation, a.k.a. Fake News, and web inaccessibility so we can all live our best lives possible.

Fake News as an area of research is relatively new and so some aspects are not very well researched, each aspect is researched separately and not intersectionally and there is no product for the user. On top of that, most of the methods used are only typically around 70% accurate, there is no method for detection fake news in languages other than English, and there is very little public documentation of their work. In terms of Web accessibility, this is an area of afterthought for developers and that makes it harder for everyone that has disabilities, has other impairments, comes from a different culture, or doesn't know English very well to actually get this wide breadth of knowledge that's out there.

At the moment, this research aims to create a functional, accessible, human-centered user interface for automatic fake news detection. The interface would include a scalable back end framework where the best methods for automatic deception detection can be added. In the future, with more time, this project hopes to be fully functional in automatically detecting fake news while being versatile, robust, and accurate.

KEYWORDS

fake news, automated deception detection, machine learning, veracity assessment, misinformation, user design, web accessibility, user interface

1 INTRODUCTION

The proliferation of "fake news", "rumors", "misinformation", "deception", "click-bait" in news media has had a resurgence partly due to things like "online social networks", "social media platforms", and "online news environments"[3, 8–16, 19–32].Every resource in the bibliography or reference list can attest to the danger of such misinformation to the public. We can see how it greatly impact the United States election, for example, or how fake tweets about an injury President Obama had around April 2013 led to instability in financial markets [30]. The spread misinformation informs public opinion and public opinion which shapes public policy, elections, how companies choose to run, the stock market/economy, views on minorities and people of other countries, and even the health and safety of the public itself. We can see the latter through the misinformation about vaccines. There is a resurgence of measles in the United States after it had been declared eradicated after January 1, 2000 due to the refusal of vaccines [17]. It impacts people in significant ways that can cause harmful and sometimes irreversible effects, of which no one is exempt. However, fake news is not the only evil at play here. As Calvo et al. mentioned in their introduction, Tim Berners-Lee stated “The power of the web is in its universality. Access by everyone regardless of disability is an essential aspect” [7], and so the lack of accessibility in the web overall is another evil the helps the spread of fake news and limits the effectiveness of any solutions out there. This is why it is necessary to fight back against the proliferation of fake news and the inaccessibility of the web.

Many other researchers have understood the necessity of understanding and combating this topic and thus, there is a large breadth of work on fake news detection. While collecting research for fake news detection, a few problems or gaps in this topic emerged. There is a lack of research in different types of media like videos and images, a general one-size fits all for all types of topic domains in a given media, the best way to deliver this solution to the people, detecting misinformation in the media about recent events, a robust solution that will not require frequent retraining, providing a solution that will work on media of different languages, and there is no consensus about the best methodology or tool to use to create this solution. The reason there are so many gaps is because this is a very large problem, which makes it hard to solve all at once in one go. The existing research all focus on certain aspects of the issues in fake news detection. Some focus on finding approaches that work with articles of all topic domains, some focus on fake video detection exclusively, and some focus on fake news detection in social media posts. If users have to go to many different sites or applications or tools to check specific pieces of media, the users will not engage with these tools because it is way too much effort. On top of those issues, there is the fact that most results from these researchers tend to be around 70% accurate and there is a lack of proper documentation about their process, tools, and data and so there's no way to even recreate their results.

While access to the web by everyone regardless of disability is essential, many researchers have noticed that the web is still not accessible. Most of the research about web accessibility was not focused on originally created solutions but rather on issues with the guidelines or evaluation tools. These papers are shedding light on how we have these guidelines and even if people are trying to meet it, it is still an afterthought in the design process and evaluation tools can't be the end all be all of web accessibility checking. At the same time there is are issues with the guidelines having the correct priorities associated with the correct issues as pointed out by disabled users and experts, as well as with the guidelines actually having all the accessibility issues that we should be paying attention to. The authors creating things for the web are mistaken on what needs to happen for their product to be accessible and would rather put style over the needs of users. There is also a large bias in the web towards making content in English with the Western culture in mind even though there are millions of non-English users that need to use the internet everyday and this just makes it harder for them to get the same information and opportunities that everyone else is.

While previous researches went out far and wide to touch all the possible aspects, this research project will reign it back in to consolidate the findings into one tool for users. The research focuses on ultimately developing a user interface that will be a robust, accurate, and versatile solution that other researches have expressed a need for [8–10, 16, 28]. The robust aspect addresses the issue of needing frequent manual retraining, the accuracy aspect addresses the issue of most automatic solutions only being able to identify fake news 70% of the time, and the versatile aspect addresses the issue that most research has been focused on specific articles. Due to time constraints, we will be focusing on the issue of how best to disseminate this service to the public in a way that will actually impact people and curb the spread of fake news. This will be the accessible user interface with a scale-able back-end framework for easily adding the machine learning scripts in the future that we create. This interface will be built from the ground up with accessibility in mind so that this work can actually reach a majority of people and attempt to stop the spread of fake news. We will attempt to address the language barrier issue and the common pitfalls of developers trying to create accessible and responsive designs. In addition to that, we will attempt to address a wider breadth of impairments than most like reading level, mobility issues, sight impairments, audio impairments, cognitive/psychological difficulties, etc. This research will also provide proper documentation for the whole process.

In the future, this solution involves extensive testing of different combinations of methods, features, and techniques for each media type to uncover the best methods, features and techniques and to achieve the highest possible accuracy. What would be produced is a combination of methods that involve machine learning and web searching that will take advantage of new knowledge being produced and to be able to grow with changes in the way news is presented. In this time frame, however, this project will be an accessible user platform that will detail what the user would see if the back-end was fully functional. It will demonstrate the best way to get most of the public to engage with this service and make it available to all regardless of any disabilities of societal impediment. It will also provide and easy framework for adding/updating the actual machine learning processes for the back-end of the framework. Another major contribution will be a responsive and accessible website and extension with multiple input methods for ease of access. The project will do everything possible to accommodate for people with impairments while still being a rich and enjoyable experience for everyone. It will be compatible with screen readers, it will have high enough color contrast between all elements on the page, it will not be higher than a fifth grade reading level, it will have the capability to be translated, it will follow common design practices, it won't fall into the common mistakes when making something accessible, etc. Most importantly, my interface will include a feedback section so that I might be able to change aspects to be the most accessible we can make them. The creation of a log of my work will also be a main part of our contribution.

This paper will be structured as follows. In Section 2, related works to this project from the literature will be discussed. Section 3 will detail the design and implementation of the solution proposed in this paper. The results from the implementation of this solution will be outlined in Section 4, and Section 5 is the conclusion.

2 RELATED WORK

In the research conducted for this proposal, the structure of related work pertaining to the automatic detection portion of the project seems to be divided into a couple of parts: Data sets, which includes the features/approaches they used to know if something is fake, Identification methods for automatic detection of those features/approaches, and occasionally there are experiment setup and results and the design of a user interface[3, 8–16, 19–32]. Another thing that a large proportion of these research articles have in common, is that they each focus on a specific and narrowed area of fake news detection. Therefore, comparing them based on what data sets and features they were working with and what identification methods they used seems more fruitful.

The related work found while doing research about web accessibility were varied in terms of what aspect of accessibility it focused on. Some focused specifically on mobile accessibility, some focused on the possible issues of evaluating accessibility [1, 5–7], some focused on the issues of the web accessibility guidelines [5, 7], some focused on detailing the different parts of the web accessibility guidelines, and more. Because of this, the research will be compared based on the issues and recommendations that it brought up. This will give us a broader view on the aspect of web accessibility.

2.1 Data Sets

Data sets are critical to creating an automatic detector of fake news because that is how one needs to test the identification and classification methods. Rubin et al. did not outline a process to detect fake news but did outline requirements for a Fake News Detection Corpus [20]. The data sets used in each article reflects the specific focus they had. For example, the data set used by Mihalcea et. al. is entire comprised of videos that were created using Amazon Mechanical Turk service and one of the data sets used by Castelo et. al. is a corpus they created themselves that includes over 14,000 political news pages drawn from 137 sites and spanning 6 years [8, 14]. If what the authors wanted to investigate was not already a well investigated area, they most likely had to create their own data set and many used a crowd-sourcing platform to do that. While the actual data sets used by authors may be very different, the features they used to identify fake news can be organized into categories.

2.1.1 Features/Approaches. When creating a tool that can recognize misinformation/deception, there needs to be something about the media that is different from truthful media,and those factors are called features. The features could be the frequency of words paired with further part-of-speech tagging [8, 10], or it could be the images associated with the news [9, 15]. Some of the features mentioned in the articles only work best with a certain type of media like images or written articles, but many can be used in conjunction with each other. The types of features have been tailored to the types of media, like click-bait, social media, or transcriptions of videos, the authors were looking at. The features do tend to fall into categories of text based or non-text based.

Table 1: Types of Features for the Types of Media

Feature	Articles/ Social Media	Topic-Agnostic	Visual/ Verbal
Image Analysis	X		X
Linked Data	X		
Social Network Behavior	X		
User Behavior Analysis	X		
Readability Features		X	
Web-Markup Features		X	
"bag of words" approach	X		X
Deep Syntax Analysis	X		
Semantic Analysis	X		
Rhetorical Structure and Discourse Analysis	X		
Lexical Analysis	X		
Syntactic and Pragmatic Analysis	X		
Extract Claims and their Dependencies	X		
Unigrams			X
Morphological Features		X	
Psychological Features		X	

Table 1 is a comparison of eleven authors and what feature they used based on what type of media they included in their research [3, 8–10, 13–15, 22, 29, 30, 32]. The authors Conroy et. al., Chen et. al., Lim et. al. are some of the authors focusing on articles and social media, the authors Castelo et. al. focused on articles but in a topic-agnostic framework specifically, and the authors Narwal et. al., Mihalcea et. al. are some of the few focusing on visual and verbal media like images and videos [8–10, 13–15]. We can see from this Table that there is a lot more variety of features used for articles and social media than the other types of media and that there are a wide range of types of features found for fake news detection. This shows that detecting falsehoods and misinformation is complex and that there are many different aspects and perspectives to consider like purely text based features like "bag of words", to user behavior analysis, to even the web-markup the media was posted on.

2.2 Identification Methods

After identifying the features that mark fake news as different from factual news, the next step is to be able to identify/classify those marks accurately. There are different ways to identify/classify ranging from manual inspection to machine learning methods. However, the proposed solution is for those who cannot recognize non-credible media or who do not have the time to discern credible from non-credible media, and so manual deception detection cannot be the only method used if used at all.

Table 2: Comparing Authors by Use of Hybrid or Non-Hybrid Approaches

Authors	Machine Learning Method	Other Type of Method
Atanasova et al. [3]	X	
Castelo et al. [8]	X	
Chen et al. [9]	X	X
Conroy et al. [10]	X	X
Lim et al. [13]	X	X
Mihalcea et al. [14]	X	
Narwal et al. [15]		X
Sharma et al. [22]	X	
Zhang et al. [29]	X	
Zhang et al. [30]		X
Zubiaga et al. [32]	X	

In Table 2, is a comparison of eleven authors on the basis of whether or not they have used a hybrid approach in identification and classification [3, 8–10, 13–15, 22, 29, 30, 32]. Two different categories of approaches has been found in these readings: machine learning methods and non-machine learning methods. From the table we can see that only three papers employed a hybrid approach using both machine learning and other types of methods. We can also see that that most authors chose to utilize machine learning methods rather than other types. Therefore machine learning should most likely be used in the overall solution. Looking a level deeper of machine learning methods, there are seven authors that use Support Vector Machines (SVM) in their research [3, 8–10, 14, 22, 32], but there are still fourteen different machine learning techniques that have been listed by the authors of these research papers. So while there seems to be a consensus on the usefulness of SVM as a method, there is still no general consensus on the best combination or overall method.

It is also important to remember that all the different methods correspond to different features and those features tend to correspond to certain types of fake news. To create something that will detect fake news in more than one medium, a mixture of identification methods/classifiers will need to be used.

2.3 Web Accessibility

Almeida and Baranauskas detail really well why we need web accessibility in their paper when they wrote, "Web accessibility is becoming an increasing concern for web-based systems that are supposed to be used by audience with diverse socio-economics differences. In scenarios of great diversity, usually found in developing countries such as China, India and Brazil, the demand for accessible solutions becomes critical," [1]. Human beings are incredibly diverse and we can't design things with only certain groups of humans in mind, but that is exactly what people are doing when creating things for the web. Many researchers have found that the web continues to have barriers to access even as we've begun to create guidelines to follow [2, 5, 7, 18]. For the purpose of this project, we will be analyzing the different issues and recommendations brought up by the research found on web accessibility.

2.3.1 Issues Dealing with Web Accessibility. One of the issues discussed in Brewer's research is fragmentation and harmonization of web accessibility standards as well as physical devices and technological capabilities [6]. Fragmentation would entail things such as different sets of standards by region or a lot of different software/technology coming up that has the same goal but different ways of processing things to get to that goal. The issue with fragmentation is that it becomes difficult to create things for the masses if everyone's devices behave differently and if everyone operates on different standards. We would have to create new training resources, new components, new content for each separate group/region and that would become an unwieldy and massive task for any developer. Harmonization is the opposite where things universally follow the same standards and processes, and the harmonization of standards is not something that is inherently a bad idea. As Brewer points out in her research, having that harmonization of standards would lead to an improvement in evaluation tools, allow the re-use of training resources for web accessibility, allow the creation of "accessible, compatible, and re-usable content" [6].

The issue with Harmonization is brought up in Prasad's work when they explain how there are segments of the user population, which include people with minimal (English) language expertise, with less high-order-thinking skills, or from a different cultural background, that will find content on the web "complex, foreign, incomprehensible and inaccessible" [18]. Harmonization brings with it an unconscious bias of creating guidelines based on your own culture and language. If we pigeon-hole our guidelines for web-accessibility we will continue to create an inaccessible for over 2 billion users whose native language is not English [18].

Brewer also discusses the myths about making things accessible which slows down the progress that could be made in creating an accessible web. One of the myths is the idea that all you need for accessibility is for all the people with disabilities to get special assistive technology to fill in the gaps of accessibility on the websites [6]. But assistive technology does not exist independently of the content on the web. If the web site lacks alt links, proper use of HTML tags, or other things, screen readers can not help those that are blind understand what is on the web site for example. This technology will not save developers from having to make their products with accessibility in mind.

The other side of that myth is the myth that Web Content Accessibility Guidelines (WCAG) 1.0 is a "stand-alone solution" [6]. Technology designers, browser and media player developers, authoring tool developers, and content developers all have complementary roles in making the web an accessible place for everyone. No one role can fix accessibility by itself and accessibility needs to be thought about when ever someone makes a product no matter what that product is if we want to be able to be accessible to all.

Some other myths about making things accessible are that text-only web sites are enough to make the web site accessible and that accessible web sites are consequently dull and boring. Having only text would help some people with certain types of visual impairment but would still make it inaccessible to those with auditory, mobility, cognitive, neurological impairments, and even for many types of visual impairments [6]. Pictures are still a necessary component to web sites and web sites can still look stylized, but "[d]ifferent disabilities have different requirements for accessibility" so there is not going to be a one-size-fits-all solution [6].

One of the main issues that even though there are guidelines that talk about how to make web content accessible, many studies show that many websites still do not conform to all the guidelines [7]. Some of the other big issues with the WCAG guidelines found were that they are not completely "machine testable" guidelines due to the fact that they are contextually, socially, and culturally dependent and that designers have difficulty in understand the consequences in accessibility [1]. That makes it very difficult for designers to actually test whether their products are accessible. Another set of accessibility guidelines are the Universal Design (UD) principles which aims "to design products and environments to be usable by everyone, to the greatest extent possible, without the need for adaptation or specialized design" [1]. The issue that arises with UD is that its focus is mainly on physical products and those guidelines are always translatable to the what is needed to support developers of web content and other non tangible products.

There have also been questions raised about the WCAG guidelines themselves like how some of the assigned priority levels of the guidelines are not congruent with the severity levels that the users who have disabilities and experts would give to the issues found, and in Calvo et al., they outline multiple studies in which they found that there are large proportions of issues reported by users with various disabilities (blind, partially sighted, dyslexic, deaf and physical impairments) that were not covered by the WCAG guidelines [7]. Because of the latter, Calvo et al. conducted audits of different types of websites and asked experts to report the accessibility issues. A total of 1214 issues were found and 6% of the issues found were issues not covered by WCAG guidelines but are known by experts as potential problems for users with disabilities [7]. Table 3 shows those issues grouped into 7 different groups and the frequency of each type of issue. This info will help greatly when creating the project because we will be aware of this issues going in and not fall into the same traps that evaluation tools might not catch.

There are tools out there that aim at helping developers check that their websites are in accordance with the WCAG guidelines. This can be very helpful in making sure that developers do not miss common accessibility errors. However, Almeida points out that some guidelines are contextually, socially, and culturally dependent and thus semi-automatic evaluation tools can fail in helping developers understand and separate which possible issues are actually problems and which are not based on the context [1]. Because context is important, it is equally important to have human inspection on these issues like making sure the alternative text actually

Table 3: Outlining the Frequency of Well-Known Issues Found by Experts That Are Not Covered by the WCAG Guidelines [7]

Issue	Frequency
Hide information incorrectly	15.8%
Do not use common design patterns	13.4%
Wide gaps between related information	8.7%
Use of custom components	39.2%
Buttons and text size are small	5.3%
Colour contrast ratio between icons and background is not enough	10.1%
Important information is not shown at the top	4.8%

corresponds to the image [7]. The downside is that many accessible solutions done consciously are based on segregating people according to have type of assistance they need which will those communities to feel more isolated and lead to content being made sub-par for certain groups because that is the easiest for the developer [1]. This will just deepen the digital divide.

Bias can also creep in when evaluating accessibility. There is the "evaluator effect" is the fact that "the observation that different evaluators in similar conditions identify substantially different sets of usability problems" [5]. There are different expertise levels in the evaluators which could affect what problems they can identify, and there is also a difficulty in reliably rating the severity of problems that they do find [5]. There could be bias caused by the sampling method adopted to select pages to evaluate, or by the error rates of the testing tools [5]. There's a lack of standardization that can lead to low reproducibility of results, which could be linked to why there is a difficulty in reproducing the results of the machine learning solutions produced by the researchers in the previous sub sections of section 2 Related Work.

2.3.2 Recommendations on Making the Web Accessible. Most of the research in this area does focus on the issues but there are some that have recommendations for how to get better accessibility in what people create. Calvo et al. described recommendations for fixing the seven groups of issues not covered by the WCAG. The solutions include not hiding interactive components visually and failing to hide them for keyboard and screen reader aides for users, to use common design patterns and common interactive elements so that users will be able to complete tasks easily, have related information shown closer to each other on the page to make the relationship visual too, make sure targets and text are big enough, assure there is a high enough color contrast ration between the background and foreground colors for every interactive component, and to show important information at the top [7]. Bailey and Gkatzidou created a lightweight model for thinking about accessibility while creating your product. They assert that we should not consider accessibility, usability, and user experience in isolation from one another because they all need to be optimal for the product to be used effectively [4]. This view has been one of the key principles of the British Standard (BS8878), but it is not apart of the WCAG. Their model also proposes three separate but interdependent and overlapping components to take in account: technical accessibility, operational accessibility, and psychological accessibility [4]. Technical accessibility are the requirements for a user to access the product/service/physical environment, which is where conformance with the guidelines and compatibility with assistive technologies comes into play [4]. Operational accessibility is how well a user can use and operate the product or navigate the physical environment once they have access to it, which refers to the efficiency, the error rate, the error recovery, and the extent the product/feature meets the users' expectations [4]. Psychological accessibility is the aspects including but not limited to how useful they find its functionality/facilities, how appropriate they are for the user, and how satisfying the overall experience is, which represents the users' desires [4].

One of the most comprehensive solutions to web accessibility is the Blueprint for a Human-Centered Safety Net by Code for America [2]. They created this human-centered safety net in the context of public benefit programs but it has a lot of relevance to this project as well. Code for America describes the human-centered safety net as something that is "simple, accessible, and easy for real people to use", "meets people where they are and provides clarity when there is confusion", and "guarantees that the needs of the clients are put first" [2]. They provide five over arching principles to abide by as well as more in-depth detail about how to successfully achieve each principle and why it matters to achieve those things.

The five principles are "Many Welcoming Doors", "Easy to Understand", "Informed Decisions", "Responsive to Changing Needs", and "Simple Actions" [2]. The Many Welcoming Doors principle's goal is to provide an "equitable and positive experience both online and in person" to make sure that people are not intimidated or barred from using the service due to access barriers [2]. The Easy to Understand principle details that users should be able to make it through the process with minimal support [2]. The Informed Decisions principle is that the users should clearly understand the implications of all of the actions they have to take throughout the process [2]. The Responsive to Changing Needs principle states that the product needs to be able to change based on the users' needs, as well as shifts in policy and budget [2]. The Simple Action principle says that each stage in the enrollment and eligibility process should be able to be completed in as few steps as possible [2]. Some of the more detailed steps with in these principles are to make the website responsive across devices, to limit the use of legal language and bold warning text (because it is intimidating), to not have a login or remote identity proofing if possible, to have an FAQ page, to have the content available in multiple languages, to have the information written at a fifth grade reading level, etc. [2].

3 DESIGN

Figure 1 details the sections/components of the overall solution. Each component is very much dependent on the processes before it but they are all separate operations. This diagram includes the user in the overall process because they are just as important to the complete process as the machine learning tool itself. The sections in this solution are

- User Input and Data Finding

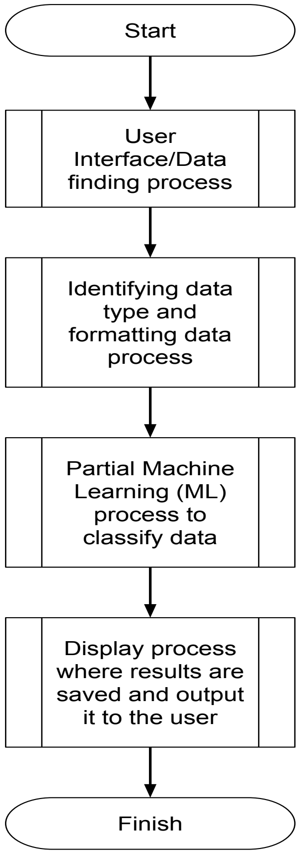

Figure 1: A High-Level, Basic Overview of the Processes in the Overall Solution

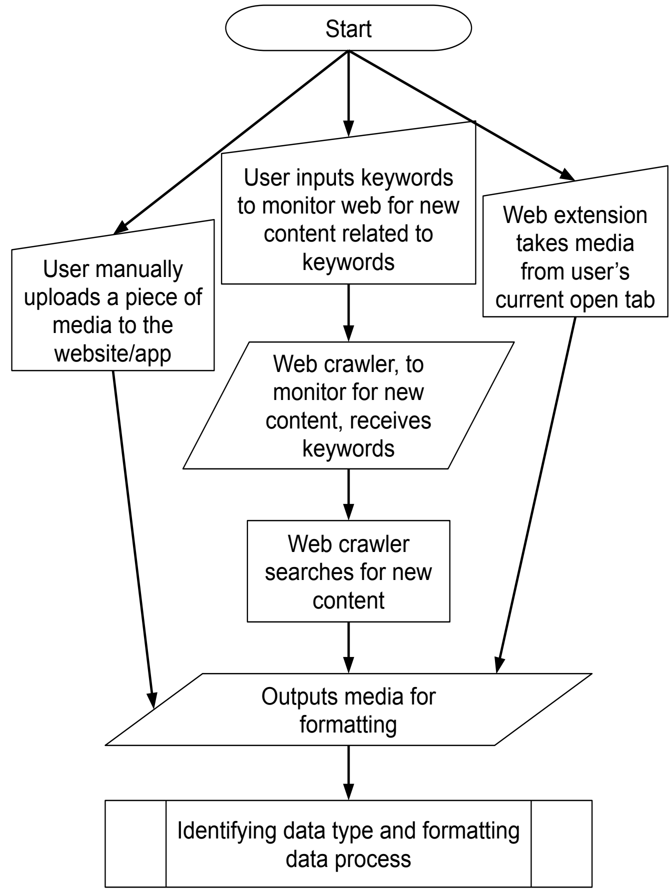

Figure 2: An Overview of the Data Finding Process and the Different Aspects of the Interface.

- Data Identifying and Formatting
- Partial Machine Learning/Back-end
- Saving and Displaying Results

3.1 User Input and Data Finding

The User Input and Data Finding Component Process is how the pieces of media to check for truthfulness will be gathered. The other part of this component is how users can interact with this fake news detector. There needs to be multiple avenues to engage with this interface to increase user-ship and affect the spread of fake news.

There are three ways in this solution a user can check for credibility in their news. They can either manually upload the files they want to check, create news alerts where input keywords to monitor the web for new content related to the keywords, install the web-extension that gets media data from their current open tab, or use any combination of these methods. With the manual upload, the user should get direct and detailed results very quickly, but this requires the most energy on the part of the user. The keyword feature would provide alerts if new media is found and display a brief synopsis of the results, which only requires minimal effort from the user to input keywords. For the keyword feature, there would need to be a web crawler that takes the keywords entered and monitor the web for content relating or matching the keywords and gather the media it finds. The last avenue is the web-extension and it will deliver the results in the form of a popup overlay on top of the current tab the user is viewing that will show the user how credible what their viewing is without any work from the user.

In this process, we would be sure that were as little steps as possible for each way to check for fake news. The extension would just require installation. The upload would require two steps: selecting the piece of media and then clicking the button to upload it. The news alerts function would require the user to login, input keywords, and press enter to receive notifications about the result. To make sure that the users are clear on how to use the fake news detector, we are going to add instructions on the page for each at the top as well as have a tutorial section with more in depth instructions as well as added pictures and videos to help show the information in different ways.

The result of this process is that there is a piece of media to check the credibility of and that will be the input for the next process.

3.2 Data Identifying and Formatting

Next in the overarching process is identifying and formatting the data we got from the User Input and Data Finding process. We are focusing on identifying what type the data is so we can properly implement the partial back-end. We want to know whether it is a picture or video or article or social media post so we can correctly match it to the data sets of already checked media until a full machine learning back-end can be implemented.The type can gathered based on the filename extension very easily.

Formatting will be important since that is how the machine learning tool will be able to see the trends in the data and learn what is credible and what is not. There will most likely need to be different formatting processes for different types of media. Images will need to have image processing tools like OCR (Optical Character Recognition) to help with formatting and Videos will need to also be transcribed to get all the features the machine learning tool needs to understand if it's credible. Translating all types of media to text is one method of checking for fake news but can also be combined with looking at raw audio features [16] or the image points itself like in the work of Steinebach et al. [24].

The formatting involves taking the media and extracting key things about it, also called features, that will help determine if it's fake or not and putting those into certain columns in a Comma Separated Values (CSV) file. At the end of this process, this formatting will output the media in a CSV file detailing its features so the machine learning tool can take it and understand it.

3.3 Partial Machine Learning/Back-end

From that point, the next process is the Machine Learning Classifying process.This component will contain the machine learning process itself but also includes the API connection and the formatting of the output. The formatting section is so that the machine learning output is in a readable format for all the possible users. Otherwise the output would only make sense to the users that have used machine learning technology before, and that is not accessible to the public. The API section is the code that will connect the machine learning tool to the front-end software and scripts such as the website/app and web-extension.

The process of classifying data is crucial to the whole solution and so it is outlined to the right in Figure 3. There is more variability in machine learning classifiers and processes but Figure 3 shows the bare bones idea behind machine learning.

The idea behind machine learning is that given enough data the machine itself can make educated decisions like a human might. Figure 3 shows the difference between training in machine learning and testing in machine learning. The training section needs large amounts of data with clear labels about what is credible and what is not so that the machine learning tool can learn to recognize what features should lead to credible rating and what features should lead to it being not credible. After training, the machine uses the knowledge it gained to classify separate singular pieces of data, which is how the solution will be used in production by the users.

The details known about the training section for this particular solution will be shown in Section 4.1 Experiment Details. This is where the data set and possible algorithms and classifiers that are used in the machine learning process will be talked about.

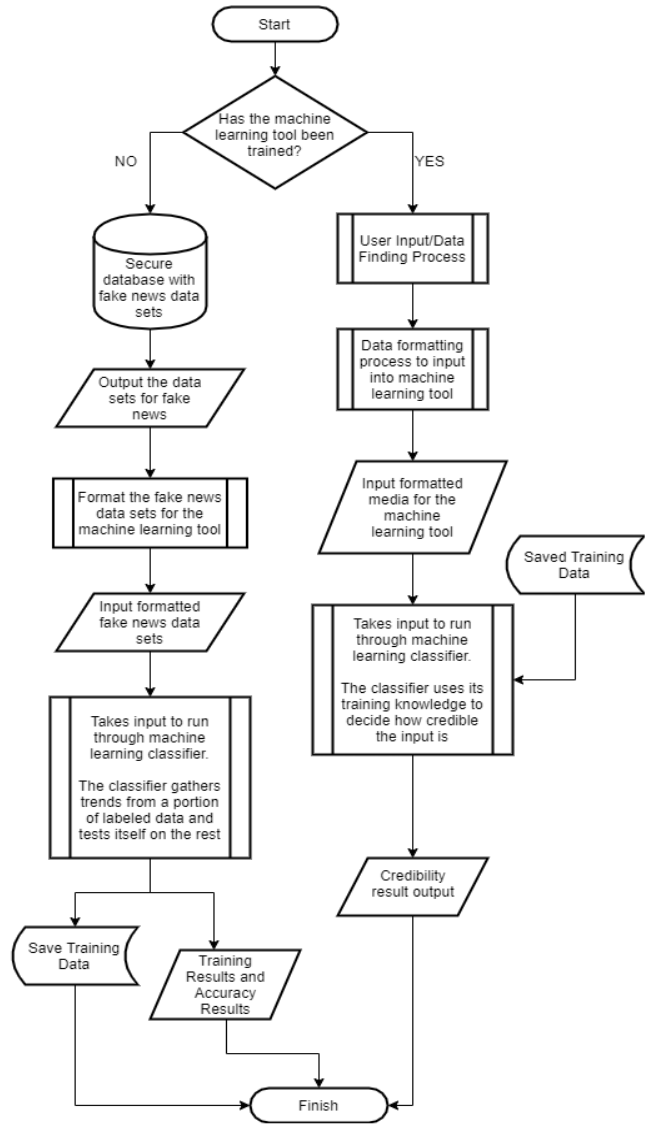

Figure 3: A High Level Overview of the Machine Learning Concept.

There does need to be a bit of a modification for the time constraint. So there will be a frame work for the machine learning scripts and back-end to be easily set up in the future and as a place holder, the formatted and identified data will be matched to the data sets we found to then be able to classify whether it is fake or not.

The output from this process would be the prediction of whether a piece of media is fake or not formatted in a way that any user could understand. The media inputted here is also passed along to the next process.

3.4 Saving and Displaying Results

After the machine learning tool classifies the inputted media as credible or not or the partial back-end identifies whether the inputted media is in the data sets collected and is credible or not, the output needs to be displayed back to the user. However, the method of display can not be the same for all the types of input as defined in section 3.1 User Input and Data Finding.

If the input came from the user as an upload, then the user would be on the website looking at the screen and so the output should be displayed on that page. If the input came from the keyword monitor list, then the user might not be on the website at all and so the user will need to be alerted of the new media and be given a brief synopsis of the results. The user can click on the notification and be taken to the website for more details. Finally, if the input came from the web-extension, then the result needs to be displayed over the current tab they are viewing in a way that catches their eye but is still out of the way, which is why it will be displayed in a popup overlay. The user can click on the overlay for a brief synopsis and can click a link on the overlay to get to the website for a more detailed explanation.

We'll need to make sure that any notifications are not popping up in a very violent or flashy way because we do not want to intimidate the user. It will scare them away from using our product. We want the notifications to be noticeable and use color to help communicate the results, with a color blindness option so that we can make sure the colors contrast and make sense to everyone. The displayed information should be in an easy to read format and we don't want to overload the users with extraneous information like how many "I's" were found in the text. At the most, we want to be able to point them to areas in the media that made the machine learning tool flag it for fake news.

At this step, the original media, the formatted media outputted by the Data Formatting process, and the results will also be saved in a PostgreSQL database. Saving it in a database allows the user to be able to revisit past results and the media in the database could be used to help detect credibility in future media by comparing it to the newly inputted media.

3.5 Interface Accessibility Design

To fully create a design that is accessible to all will need a lot of iteration and research, but we can start with a base of things we know to help accessibility. Our base will include adhering to the Web Content Accessibility 2.1 Guidelines, and the Code for America Blueprint for a Human Centered Safety Net. We will start by including such things as an easy login and sign up process, a resources tab with FAQ, Feedback, Tutorial, and Similar Services pages, compatibility with screen readers and other aides, content that is simple, clear, and concise at a 5th grade reading level, and simple, clear layouts that are following common design practices.

4 RESULTS

This solution is in part an engineering project and part a research project. we need to both create a product and research certain elements to create an exact methodology and process. Because of that, the results section will be fleshed out continually as new data comes in and the learning process restarts to accommodate the new things learned. In this section, we will detail my design for the experiments needed, the process for the experiments, and the outcomes of those experiments.

4.1 Experiment Design

For this solution, We've started by adhering to the WCAG and the Code For America Blueprint as well as following the principles for reusable code. We've included the experiment we laid out for the machine learning detection portion even though the results were inconclusive because we hope to continue this project and create a better experiment in the future. We have also included the experiment for user testing that will be used in the future as my product was not ready in time.

4.1.1 Machine Learning Experiment. This experiment will essentially be training and testing the machine learning tool over and over with different combinations and comparing the accuracy of each round.

First, we'll need to collect the various data sets that authors have used in their respective papers. If this is possible, then there will be no need for us to do data collection since there should already be a variety of media types and topic domains and types of articles in those data sets. They would have all been verified by credible sources. The data sets collected would then be compiled into one master data set that the machine learning tool could be trained on.

The next part of the experiment would be testing machine learning tool. An automatic loop will be set up where a subset of features are chosen and a subset of algorithms are chosen to test the data set, the results are recorded in a spreadsheet, and then the next subset of both is chosen, those results are recorded, and so on and so forth.

After all the possibilities have been tested, the results on the accuracy for each combination will be compared and the most accurate solution will be chosen for this project.

To illustrate what some options are, Table 3 below and Table 4 on the next page show the identification methods and types of classifiers that some of the authors have used in their research. There are more than just these options to look for which is why there will be around 3 weeks of testing.

4.1.2 User Testing Experiment. This experiment will be used to find the best way to design a user interface like website.

The first part of the experiment will be finding research on this subject and combining that with the knowledge we have learned in the past about web and software development. we will need to create a Minimum Viable Product (MVP) for the users to test.

Then we will create a testing group to use the interface and give us feedback about the usability and functionality. The testing group would need to involve people of different demographics and also of people with different political beliefs because we want this to be effective towards everyone. Any one must come to trust the results from this solution and we don't want it to be skewed or biased towards one group or another. The process for finding a testing group will have to start off with the people that we can reach through social media and hopefully from there it can gain traction and I can reach more people beyond my followers.

Table 4: Types of Identification Methods/Classifiers Using Machine Learning

Method/Classifier(Machine Learning)	Authors
Support Vector Machines (SVM)	[3, 8–10, 14, 22, 32]
Naive Bayesian	[9, 10, 14]
Neural Network Analysis	[9]
Adaboost	[13]
Gradient Boosting Tree	[13]
Random Forest	[8, 13]
Extremely Randomized Trees	[13]
K-Nearest Neighbors (KNN)	[8]
Radial Basis Function Kernel (RBF)	[3]
bi-LSTM Recurrent Neural Network	[3]
Convolutional Neural Networks (CNN)	[22]
Logistic Regression	[22]
Recurrent Neural Network (RNN)	[22]
Vector Space Model (VSM)	[10]

Table 5: Types of Identification Methods/Classifiers (Not Machine Learning)

Method/Classifier	Authors
Frequency Analysis	[9]
Probability Context Free Grammars (PCFG)	[9, 10]
Image Detection	[9]
Image Caption Analysis	[9]
Web Traffic Analysis	[9]
Web Metadata Analysis	[9]
Random Selection (Rand)	[30]
High Degree (HD)	[30]
PageRank (PR)	[30]
Minimum Monitor Set Construction (MMSC) Algorithm	[29]
User/manual verification	[13, 15]

The questions for the testing group will change for each iteration depending on what we needed to fix and change from the last testing group. The questions for the first testing group have not been created yet because we do not have a MVP to test yet.

This will be an iterative process where we take the feedback given and update the interface to take into account what we can and then a new round of testing will take place. Because this was unable to happen before May, this will be a continuing process that only stops when there is nothing we could possibly improve upon.

4.2 Progress Made on Product and Experiments

So far, we have collected data sets and tried to replicate previous work to make sure that the tools we have are functional and their work is something I can base mine off of. We have also begun to create the Minimum Viable Product. Below we will outline the general process for each thing we've done and will link to a full documentation of steps and troubleshooting issues and fixes after the project is fully completed.

4.2.1 Data Set Collection. To be able to save time for this project, we decided to not create our own fake news data sets and find the data sets listed in the research papers read. Most papers did not list a link to the data set and not every researcher had a personal site or a Github with the data set posted. Therefore we had to do some intense google searching with clues from the papers to locate the data sets. In the end we have found about 27 different verified fake news data sets.

4.2.2 Recreation of Previous Work. This has been a stumbling block for this project seeing that there is not enough documentation on the process taken by other researchers. We had to make assumptions about tools used for some of the researchers by researching ourselves about machine learning tools or taking listed tools in other projects and applying them to these papers. Thus we began to use Weka as a tool to test the data sets with a different range of classifiers and features.

Upon further experimentation, we have concluded that Weka could not help us create tests to find what combination of features and identification methods work the best in this context. In fact, we found that we would not have the time to be able to reproduce anyone's work because there was not enough information made public to other researchers like us. This is why we had to go a different route but we felt that this stumbling block is important to note due to the importance of reproducibility in science to be able to confirm findings as valid. You can find more info about the process we took to figure out the work was not reproducible in the time frame in our documentation posted on Github, which will be linked when finished.

4.3 Results of Minimum Viable Product

What we've found so far is that our empty framework of a website has no discernible errors except two contrast errors for a credit below an image according to the automatic evaluators and my attempts to check for any issues. We've tested two separate types of evaluators so far because they are the only ones that will test your website if it is still in development, and we've found that the earlham.edu site has 14 errors and 17 contrast errors on the homepage alone and that the portfolios.cs.earlham.edu site has 1 error, 916 contrast errors, and 129 alerts. So comparatively our project is doing well accessibility wise and we are continuing to work on developing content for the website, developing the extension, and setting up the back-end framework.

5 CONCLUSION

Our goal by the end of this process was to create a functional user interface for automatic fake news detection that is accessible and human-centered, and that would include a scale-able back-end framework where the best methods for automatic deception detection can be added. At this time we have created an interface framework that has accessibility built in but the back-end is not connected to the website yet but is in progress.

There are several points that we would like to continue developing in the future because this is a full-stack and very large project. In terms of the back-end, we would want to take the time to re-create the machine learning solutions of other researchers and begin the work of integrating them in to the interface with credit and then improving upon them. The interface would be updated once new changes that don't break everything are found. We would also like to do more research in what server would be the best place to hold this project and look into detecting deep fakes, fake news in other languages, finding exactly what sections of the media we test are truthful and what were deemed not credible. In terms of the front-end, we would like to create better graphics and palettes to give the user a better design and aesthetic as well as be able to show the users more details about why the results came back the way it did. We would also like to create add-ons for social media apps and sites for more direct action in stopping the spread of misinformation. In terms of the code itself, we would like to spend more time researching react UI libraries and creating our own components for things that are very widely used on our interfaces so that it can blend in more seamlessly to our design and palette as well as be highly functional.

ACKNOWLEDGMENTS

Thank you to Dr. Charlie Peck, Dr. Xunfei Jiang, Dr. Rebecca Thomas, and the whole CS department to listening to my ideas and helping me refine them and guide me through this process. Thank you to Kharis Murphy for always being by my side and helping me work through the issues of this idea. Thank you to Lillian Gray and Laurence Ruberl who are in the same boat and provide support and feedback for this process.

REFERENCES

[1] Leonelo Dell Anhol Almeida and Maria Cecília Calani Baranauskas. 2010. Universal design principles combined with web accessibility guidelines: a case study. In *Proceedings of the IX Symposium on Human Factors in Computing Systems (IHC '10)*. Brazilian Computer Society, Belo Horizonte, Minas Gerais, Brazil, 169–178.

[2] Code for America. [n.d.]. Blueprint for a Human-Centered Safety Net / Code for America. https://www.codeforamerica.org/safetynetblueprint Library Catalog: www.codeforamerica.org.

[3] Pepa Atanasova, Preslav Nakov, Lluís Màrquez, Alberto Barrón-Cedeño, Georgi Karadzhov, Tsvetomila Mihaylova, Mitra Mohtarami, and James Glass. 2019. Automatic Fact-Checking Using Context and Discourse Information. *Journal of Data and Information Quality* 11, 3 (May 2019), 1–27. https://doi.org/10.1145/3297722

[4] Chris Bailey and Voula Gkatzidou. 2017. Considerations for Implementing a Holistic Organisational Approach to Accessibility. In *Proceedings of the 14th Web for All Conference on The Future of Accessible Work (W4A '17)*. Association for Computing Machinery, Perth, Western Australia, Australia, 1–4. https://doi.org/10.1145/3058555.3058571

[5] Giorgio Brajnik. 2009. Validity and reliability of web accessibility guidelines. In *Proceedings of the 11th international ACM SIGACCESS conference on Computers and accessibility (Assets '09)*. Association for Computing Machinery, Pittsburgh, Pennsylvania, USA, 131–138. https://doi.org/10.1145/1639642.1639666

[6] Judy Brewer. 2004. Web accessibility highlights and trends. In *Proceedings of the 2004 international cross-disciplinary workshop on Web accessibility (W4A) (W4A '04)*. Association for Computing Machinery, New York City, New York, USA, 51–55. https://doi.org/10.1145/990657.990667

[7] Rocio Calvo, Faezeh Seyedarabi, and Andreas Savva. 2016. Beyond Web Content Accessibility Guidelines: Expert Accessibility Reviews. In *Proceedings of the 7th International Conference on Software Development and Technologies for Enhancing Accessibility and Fighting Info-exclusion (DSAI 2016)*. Association for Computing Machinery, Vila Real, Portugal, 77–84. https://doi.org/10.1145/3019943.3019955

[8] Sonia Castelo, Thais Almeida, Anas Elghafari, Aécio Santos, Kien Pham, Eduardo Nakamura, and Juliana Freire. 2019. A Topic-Agnostic Approach for Identifying Fake News Pages. In *Companion Proceedings of The 2019 World Wide Web Conference on - WWW '19*. ACM Press, San Francisco, USA, 975–980. https://doi.org/10.1145/3308560.3316739

[9] Yimin Chen, Niall J. Conroy, and Victoria L. Rubin. 2015. Misleading Online Content: Recognizing Clickbait as ?False News?. In *Proceedings of the 2015 ACM on Workshop on Multimodal Deception Detection - WMDD '15*. ACM Press, Seattle, Washington, USA, 15–19. https://doi.org/10.1145/2823465.2823467

[10] Niall J. Conroy, Victoria L. Rubin, and Yimin Chen. 2015. Automatic deception detection: Methods for finding fake news: Automatic Deception Detection: Methods for Finding Fake News. *Proceedings of the Association for Information Science and Technology* 52, 1 (2015), 1–4. https://doi.org/10.1002/pra2.2015.145052010082

[11] Rui Hou, Veronica Perez-Rosas, Stacy Loeb, and Rada Mihalcea. 2019. Towards Automatic Detection of Misinformation in Online Medical Videos. In *2019 International Conference on Multimodal Interaction on - ICMI '19*. ACM Press, Suzhou, China, 235–243. https://doi.org/10.1145/3340555.3353763

[12] Alireza Karduni, Isaac Cho, Ryan Wesslen, Sashank Santhanam, Svitlana Volkova, Dustin L Arendt, Samira Shaikh, and Wenwen Dou. 2019. Vulnerable to misinformation?: Verifi!. In *Proceedings of the 24th International Conference on Intelligent User Interfaces - IUI '19*. ACM Press, Marina del Ray, California, 312–323. https://doi.org/10.1145/3301275.3302320

[13] Wee Yong Lim, Mong Li Lee, and Wynne Hsu. 2017. iFACT: An Interactive Framework to Assess Claims from Tweets. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management - CIKM '17*. ACM Press, Singapore, Singapore, 787–796. https://doi.org/10.1145/3132847.3132995

[14] Rada Mihalcea, Verónica Pérez-Rosas, and Mihai Burzo. 2013. Automatic detection of deceit in verbal communication. In *Proceedings of the 15th ACM on International conference on multimodal interaction - ICMI '13*. ACM Press, Sydney, Australia, 131–134. https://doi.org/10.1145/2522848.2522888

[15] Vishwajeet Narwal, Mohamed Hashim Salih, Jose Angel Lopez, Angel Ortega, John O'Donovan, Tobias Höllerer, and Saiph Savage. 2017. Automated Assistants to Identify and Prompt Action on Visual News Bias. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems - CHI EA '17*. ACM Press, Denver, Colorado, USA, 2796–2801. https://doi.org/10.1145/3027063.3053227

[16] Demetris Paschalides, Alexandros Kornilakis, Chrysovalantis Christodoulou, Rafael Andreou, George Pallis, Marios Dikaiakos, and Evangelos Markatos. 2019. Check-It: A plugin for Detecting and Reducing the Spread of Fake News and Misinformation on the Web. In *IEEE/WIC/ACM International Conference on Web Intelligence on - WI '19*. ACM Press, Thessaloniki, Greece, 298–302. https://doi.org/10.1145/3350546.3352534

[17] Varun K. Phadke, Robert A. Bednarczyk, Daniel A. Salmon, and Saad B. Omer. 2016. Association Between Vaccine Refusal and Vaccine-Preventable Diseases in the United States. *JAMA* 315, 11 (March 2016), 1149–1158. https://doi.org/10.1001/jama.2016.1353

[18] Gollapudi VRJ Sai Prasad. 2017. Renarrating Web Content to Increase Web Accessibility. In *Proceedings of the 10th International Conference on Theory and Practice of Electronic Governance (ICEGOV '17)*. Association for Computing Machinery, New Delhi AA, India, 598–601. https://doi.org/10.1145/3047273.3047292

[19] Julio C. S. Reis, Andre Correia, Fabricio Murai, Adriano Veloso, Fabricio Benevenuto, and Erik Cambria. 2019. Supervised Learning for Fake News Detection. *IEEE Intelligent Systems* 34, 2 (March 2019), 76–81. https://doi.org/10.1109/MIS.2019.2899143

[20] Victoria L. Rubin, Yimin Chen, and Niall J. Conroy. 2015. Deception detection for news: Three types of fakes. *Proceedings of the Association for Information Science and Technology* 52, 1 (2015), 1–4. https://doi.org/10.1002/pra2.2015.145052010083

[21] Diego Saez-Trumper. 2014. Fake tweet buster: a webtool to identify users promoting fake news ontwitter. In *Proceedings of the 25th ACM conference on Hypertext and social media - HT '14*. ACM Press, Santiago, Chile, 316–317. https://doi.org/10.1145/2631775.2631786

[22] Karishma Sharma, Feng Qian, He Jiang, Natali Ruchansky, Ming Zhang, and Yan Liu. 2019. Combating Fake News: A Survey on Identification and Mitigation Techniques. *ACM Transactions on Intelligent Systems and Technology* 10, 3 (April 2019), 1–42. https://doi.org/10.1145/3305260

[23] Maniz Shrestha. 2018. Detecting Fake News with Sentiment Analysis and Network Metadata. (2018), 6.

[24] Martin Steinebach, Karol Gotkowski, and Hujian Liu. 2019. Fake News Detection by Image Montage Recognition. In *Proceedings of the 14th International Conference on Availability, Reliability and Security - ARES '19*. ACM Press, Canterbury, CA, United Kingdom, 1–9. https://doi.org/10.1145/3339252.3341487

[25] Michela Del Vicario, Walter Quattrociocchi, Antonio Scala, and Fabiana Zollo. 2019. Polarization and Fake News: Early Warning of Potential Misinformation Targets. *ACM Transactions on the Web* 13, 2 (March 2019), 1–22. https://doi.org/10.1145/3316809

[26] Soroush Vosoughi, Mostafa 'Neo' Mohsenvand, and Deb Roy. 2017. Rumor Gauge: Predicting the Veracity of Rumors on Twitter. *ACM Transactions on Knowledge Discovery from Data* 11, 4 (July 2017), 1–36. https://doi.org/10.1145/3070644

[27] Yaqing Wang, Fenglong Ma, Zhiwei Jin, Ye Yuan, Guangxu Xun, Kishlay Jha, Lu Su, and Jing Gao. 2018. EANN: Event Adversarial Neural Networks for Multi-Modal Fake News Detection. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining - KDD '18*. ACM Press, London, United Kingdom, 849–857. https://doi.org/10.1145/3219819.3219903

[28] Fan Yang, Shiva K. Pentyala, Sina Mohseni, Mengnan Du, Hao Yuan, Rhema Linder, Eric D. Ragan, Shuiwang Ji, and Xia (Ben) Hu. 2019. XFake: Explainable Fake News Detector with Visualizations. In *The World Wide Web Conference on - WWW '19*. ACM Press, San Francisco, CA, USA, 3600–3604. https://doi.org/10.1145/3308558.3314119

[29] Huiling Zhang, Md Abdul Alim, Xiang Li, My T. Thai, and Hien T. Nguyen. 2016. Misinformation in Online Social Networks: Detect Them All with a Limited Budget. *ACM Transactions on Information Systems* 34, 3 (April 2016), 1–24. https://doi.org/10.1145/2885494

[30] Huiling Zhang, Alan Kuhnle, Huiyuan Zhang, and My T. Thai. 2016. Detecting misinformation in online social networks before it is too late. In *2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*. IEEE, San Francisco, CA, USA, 541–548. https://doi.org/10.1109/ASONAM.2016.7752288

[31] Qiang Zhang, Aldo Lipani, Shangsong Liang, and Emine Yilmaz. 2019. Reply-Aided Detection of Misinformation via Bayesian Deep Learning. In *The World Wide Web Conference on - WWW '19*. ACM Press, San Francisco, CA, USA, 2333–2343. https://doi.org/10.1145/3308558.3313718

[32] Arkaitz Zubiaga, Ahmet Aker, Kalina Bontcheva, Maria Liakata, and Rob Procter. 2018. Detection and Resolution of Rumours in Social Media: A Survey. *Comput. Surveys* 51, 2 (Feb. 2018), 1–36. https://doi.org/10.1145/3161603