

Veritas Annotator: Discovering the Origin of a Rumour

Lucas Azevedo

Insight Centre for Data Analytics

lucas.azevedo

@insight-centre.org

Mohamed Moustafa

National University of Ireland, Galway

m.moustafal

@nuigalway.ie

Abstract

Defined as the intentional or unintentional spread of false information (K et al., 2019) through context and/or content manipulation, fake news has become one of the most serious problems associated with online information (Waldrop, 2017). Consequently, it comes as no surprise that Fake News Detection has become one of the major foci of various fields of machine learning and while machine learning models have allowed individuals and companies to automate decision-based processes that were once thought to be only doable by humans, it is no secret that the real-life applications of such models are not viable without the existence of an adequate training dataset. In this paper we describe the Veritas Annotator, a web application for manually identifying the origin of a rumour. These rumours, often referred as claims, were previously checked for validity by Fact-Checking Agencies.

1 Introduction

”As an increasing amount of our lives is spent interacting online through social media platforms, more and more people tend to seek out and consume news from social media rather than traditional news organizations.” (Shu et al., 2017). This change in societal behaviour has made it much easier for some malicious authors to confuse the public opinion through lies and deception. Articles, tweets, blog posts, and other media used for spreading fake news usually include URLs(Uniform Resource Locators) to fake news websites that are often heavily biased or satirical in nature. Such content is created either for propaganda and political attacks (Waldrop, 2017), or for entertainment purposes by the infamous ”trolls”, individuals who aim to disrupt communication and influence consumers into emotional distress.

To better understand the necessity of improvements in the automatic fact-checking field, add to the above described scenario the fact that when it comes to identifying a false claim, we, humans cannot perform a simple binary classification over deceptive statements with an accuracy much better than chance, In fact, “just 4% better, based on a meta-analysis of more than 200 experiments.” (Bond Jr and DePaulo, 2006) and typically find only one-third of text-based deceptions (George and Keane, 2006; Hancock et al., 2004). This reflects the so-called ‘truth bias’ or the notion that people are more apt to judge communications as truthful (Vrij, 2000).

Fortunately, there are a number of Fact Checking (FC) agencies such as Snopes¹, Full Fact², Politifact³, Truth or Fiction⁴, etc. where journalists work on the hard tasks of: monitoring social media, identifying potential false claims and debunking or confirming them (Babakar and Moy, 2016), while providing a narrative that includes sources related to that claim. Those sources are mainly included in the text in the form of URLs and could be any type of web document that refers to the rumour being checked, debunking it or supporting. In this article, we use the term origin to refer to any supporting source. Despite the constant effort of the FC agencies, manual fact checking is an intellectually demanding and laborious process, and as Jonathan Swift once said in his classic essay “the Art of political lying”: “Falsehood flies, and truth comes limping after it” (Arbuthnot and Swift, 1874).

In this scenario, the creation of a fast, reliable and automatic way of detecting fake news (Adair et al., 2017) being spread on the internet is of the

¹<https://www.snopes.com>

²<https://fullfact.org>

³<https://www.politifact.com>

⁴<https://www.truthorfiction.com>

utmost need.

2 Motivation

Different types of modalities exist when it comes to automatic fake news detection in text (Azevedo, 2018; K et al., 2019). Here we will group them in accordance to the nature of data they take as input: social network based, where indicators as user statistics, propagation structure and behaviour of the network are considered as features; content based, where the content is what is analyzed, whether linguistic, psycho-linguistic, statistical, stylometric or a mix of those are taken into account. The Veritas (VERIfying Textual ASpects) Dataset initiative intends to improve classifiers that fall into this category; and temporal based, where a correlation between timestamps of users, events and/or articles and the genuineness of a web-document is created.

In order to improve the efficiency of content based classifiers, the retrieval of the entire origin text is essential for training a deep learning model, since the larger the text body retrieved the higher the likelihood of obtaining good measurements for the considered linguistic features. Focusing solely on microblogs, such as Twitter, has been avoided as not only their average text's length would not fit the linguistic approach but also most of them contain urls and/or images that do not convey semantic information or cannot be processed by our textual approach, respectively. The ultimate goal of our work is to develop such classifier, but in this article we will present the journey on the initial step: the dataset creation process. After having a sufficiently large dataset, that includes the origins of the checked claims, certain linguistic and stylometric features can be extracted from them and used to train the our goal model.

3 State of the Art

3.1 Available Corpora on Fake News

The lack of suitable corpora for the intended approach is the main influence behind the creation of the Veritas Dataset, and by consequence, the Veritas Annotator. Below we present a list of datasets commonly used in related tasks. Note that, although those are valuable resources for many related tasks, none of them include the three most important characteristics required for a content based supervised classifier: high volume of en-

tries, gold standard labels and the fake news articles (i.e., the origin) on their whole.

Emergent is a data-set created using the homonymous website as source, a digital journalism project for rumour debunking containing 300 rumoured claims and 2,595 associated news articles - a counterpart to named 'source article' in Veritas Dataset. Each claim's veracity is estimated by journalists after they have judged that enough evidence has been collected (Ferreira and Vlachos, 2016). Besides the claim labeling, each associated article is summarized into a headline and also labelled regarding its stance towards the claim.

NECO 2017 is an ensemble of three different datasets (Horne and Adali, 2017), summing up 110 fake news articles, more than 4k real stories and 233 satire stories. While the datasets listed above can prove useful for certain purposes, their low number of fake news entries make them insufficient for properly training a classification model.

FakeNewsNet is a data repository containing a collection of around 22K real and fake news obtained from Politifact and GossipCop⁵ FC websites. Each row contains an ID, URL, title, and a list of tweets that shared the URL. It also includes linguistic, visual, social, and spatiotemporal context regarding the articles. This repository could still be used for supervised learning models if it were not for the fact that it doesn't provide sufficiently long texts to be used by a classifier based on linguistic aspects. For the same reason, CREDBANK (Mitra and Gilbert, 2015) and PHEME (Derczynski and Bontcheva, 2014) are also unsuitable the authors' use case. Those three datasets focus on the network indicators (e.g. number of retweets, sharing patterns, etc) of fake news, instead of its contents. CREDBANK is a crowd sourced corpus of "more than 60 million tweets grouped into 1049 real-world events, each annotated by 30 human annotators", while PHEME includes 4842 tweets, in the form of 330 threads, related to 9 events.

⁵<https://www.gossipcop.com>

NELA2017 is a large news article collection consisting of 136k articles from 92 sources created for studying misinformation in news articles (Horne et al., 2018). Along with the news articles, the dataset includes a rich set of natural language features on each news article, and the corresponding Facebook engagement statistics. Unfortunately, the dataset does not include labels regarding the veracity of each article.

BuzzFeed-Webis 2016 includes posts and linked articles shared by nine hyperpartisan publishers in a week close to the 2016 US elections. All posts are fact-checked by journalists from BuzzFeed. The dataset contains more than 1.6K articles which are labeled using the scale: no factual content, mostly false, mixture of true and false, and mostly true. Regrettably, the author obtained poor results on detecting fake news with this data, while managing to discriminate between hyperpartisan and mainstream articles (Potthast et al., 2018).

LIAR is another corpus used for training models on fake news detection. It includes around 13K human-labeled short statements which are rated by the fact-checking website PolitiFact into labels for truthfulness using the scale: pants-fire, false, barely-true, half-true, mostly-true, and true (Wang, 2017). The domain-restricted data as well as the small amount of text that can be retrieved from this corpus makes it unsuitable for linguistic fake news detection for generic domains.

Another large volume fake news dataset was created by scraping text and metadata from 244 websites tagged as “bullshit” by the BS Detector Chrome Extension. However, it is not a gold standard dataset as the scraped data was not manually verified.

3.2 Related Work

Other work has been done to identify the origin of rumors/fake claims. In (Popat et al., 2018), Popat et al. have used the entities present on the article headline to find possible origins on search engines. Wang et al., from Google, have also presented a similar approach to the problem with the addition of click-graph queries (Wang et al., 2018), that re-

turn information about which link was clicked by the users after a query was made.

FANE (Rehm et al., 2018) would be the work considered the most similar to ours. It proposes a set of webpages annotations, automatic and manual, that could make the user aware of the veracity of that page’s content. The article presents a somewhat abstract idea of implementation and makes clear that the approach would only be effective when the browsers and content vendors adopt the web annotation standards proposed by W3C. Nonetheless, we fully agree with the authors when they state that human input is imperative if we want to win the battle against misinformation.

In some applications, the origin identification task can be similar to stance classification, which was the target task for the FNC-1 challenge, where obtained the best results with a combination between a deep learning model and a boosted tree classifier. Although there is no publication describing the classifier, this blogpost⁶ explains their approach.

4 Creating our Dataset

With the requirements for a linguistic-based classifier described in the last section in mind, how could a dataset that would include not only a manually verified label over the veracity of a claim, but also the web article from where that claim could be extracted? It was decided to divide the process into two steps:

4.1 Crawling fact-checking articles

We have been able to collect about 11.5 thousand origin candidates from more than 6 thousand fact checking (FC) articles by using specific scripts for each fact-checking agency and with the aid of various third-party libraries as newspaper3k⁷, beautifulsoup⁸, scrapy⁹, depending on the structure of the website. Each one of those articles include a claim that was checked by a journalist, i.e., the article’s author and a verdict regarding the claim’s veracity. Along with the claim there is a narrative where the author explains how the many sources were used to come to the final verdict. In most of the times, one (or more) of the sources is also an

⁶<https://blog.talosintelligence.com/2017/06/talos-fake-news-challenge.html>

⁷<https://pypi.org/project/newspaper3k/>

⁸<https://www.crummy.com/software/BeautifulSoup/bs4/doc/>

⁹<https://scrapy.org>

origin of the checked claim. Here we define the origin of a claim as a source that directly supports the claim.

At this stage, each of the FC articles were represented by an entry in our database, with the following attributes:

Page The the FC Article URL.

Claim The main checked claim, often included in both the FC article's and the Claim Origin's headline.

Claim Label The verdict provided by the journalist over the main claim. This label regards how much of truth the journalist found in the claim. Different agencies have different label sets but they mainly vary from truth to false, including intermediate values and one or more labels to address claims that could not be checked neither debunked.

Tags The set of tags assigned to the claim by the fact-checking agency. They are similar to hashtags¹⁰ on twitter and describe abstractly the topic of the claim and the entities cited by it.

Date The date the claim was checked. More precisely, the publishing date of the FC Article identified by page. To obtain this attribute, we make use of the public available service provided by (SalahEldeen and Nelson, 2013). This interface, makes use of search engines' indexing, as well as HTTP header and foot stamps in `archive.is` and twitter. If that approach doesn't work, newspaper3k is used.

Author The journalist that signs the FC article.

Source list A list of source URLs contained in the FC article, including the possible origin(s).

4.2 Identifying the origin amongst the sources.

Following the acquisition of the FC articles it was still needed to identify the claim's origin from amongst the list of URLs mentioned by the FC article, i.e., the sources. The actual complexity of this task surpassed our initial expectations. Many

different approaches have been applied and evaluated, always following the same process of manually checking a representative sample of the selected origins. On each evaluation, the sample size was defined in order to have 95% confidence level with 5% confidence interval. Here we explain briefly the different approaches tried:

At first, it was assumed that the text contained in the first `<blockquote>` HTML tag would be the origin. That assumption was correct on 74% percent of the time but since only the content of the first `<blockquote>` tag was considered, there were many cases where that was only partially the origin's content. If, instead, every `<blockquote>` content was assumed to be from origins, there would be cases where snippets from multiple origins would be mixed, or in even worse scenarios, the inclusion of textual content from non-origin sources. Adding this level of noise to the data would make the training of a classifier unfeasible.

The approach was then changed to assuming that the first link on the FC article was generally the origin. This was correct only on 53% of the samples analyzed.

Having failed on the first two attempts to correctly identify the origin of the claim checked, we were determined to try another heuristic, this time making use of a stance classification ensemble model¹¹, that would consider all the sources from a given FA, obtain their contents, and calculate the agreement score between the FA article's claim and the sources' contents by a linear combination of a convolutional network and a gradient boosted tree classifier. For each FC article, the source with the highest score would be then considered the origin. This worked really well in the cases where there is an origin amongst the sources, but since those do not represent the totality of the samples, the overall accuracy of the approach was lower than expected.

We then had to resort to manual annotation, detailed in the section below. In summary, by the above mentioned experiences on the origin identification task, we could define some simple filtering rules that restrict the list of origin candidates for each FC article, the remaining OCs are then presented to the user annotating, who is asked to vote on whether the current OC is indeed an origin or not.

¹⁰<https://en.wikipedia.org/wiki/Hashtag>

¹¹<https://github.com/Cisco-Talos/fnc-1>

Ver.	#Entries	Attributes	#Samples	Correct Origin	Agencies
1.0	4663	FC Article, Claim, Origin Text, Label	355	74%(partial)	Snopes.com
2.0	5107	FC Article, Claim, Label, Article Date, Author, Tags, Origin Text, Origin Domain, Origin URL	357	53%(whole)	Snopes.com Politifact.com Emergent.info
2.1	6671	FC Article, Claim, Label, Article Date, Author, Tags, Origin Text, Origin Domain, Origin URL	363	47%(whole)	Snopes.com Politifact.com FactCheck.org Emergent.info

Table 1: Versions of Veritas

By having human indicating what are the origins of each claim, not only the suitable data collection for our Automatic Fake News Classifier is generated, but the very task of origin identification can be, at this point, automated by training another classification model that would also incorporate the simple filtering rules we have defined and, in a circular manner, learn to identify more origins, or at least, better origin candidates.

Table ?? shows the number of entries of the Veritas Dataset¹² on each stage, since the FC article crawling step is executed periodically, the total number of entries changes as new pages were introduced. On the other hand, more refined filtering rules were implemented and some entries included in past versions were removed in the subsequent ones.

It is important to note that since each FC article can contain any number of sources, the first attribute of the dataset (FC article URL) is not unique on each entry, at this stage.

By the end of the origin identification process, instead of having a source list for each entry of our dataset, only the identified origin URL will remain, along with some of it’s attributes:

Origin URL The URL referring to the web-page that originated the claim.

Origin Domain The Origin URL’s domain. This can have great impacts in results of a neural network classifier accuracy, or even in the weighting of a simpler classifier method. Examples of using source rank based on the URL domain as a cue for its veracity are

not new (Popat et al., 2017; Nakashole and Mitchell, 2014).

Origin Text The whole text extracted from the Origin URL, from where the linguistic aspects could be measured and used as features by a classifier.

Origin Date Similar to the above described FC article date.

If a FC page did not have any of it’s sources identified as an origin, it will not be included in the filtered version of the dataset.

5 The Annotation Process

5.1 Task and terms definitions

Given a claim (a statement) checked by a FC Agency article (e.g. snopes, politifact, truthorfiction, etc.) and a source contained in that article, i.e., an origin candidate (OC), the task consists in deciding whether or not the source could be considered the origin of the Claim. As defined earlier, an origin is a source that directly supports the claim. More specifically, in order to be considered an origin:

- It should support what is being stated in the claim, not necessarily with the exact same words.
- It has to be more than just related.
- Directly here means it should not simply repeat or proxy other articles supporting or denying the claim.
- It doesn’t has to be the first document to publicize that claim.

¹²<https://github.com/lucas0/VeritasCorpus>

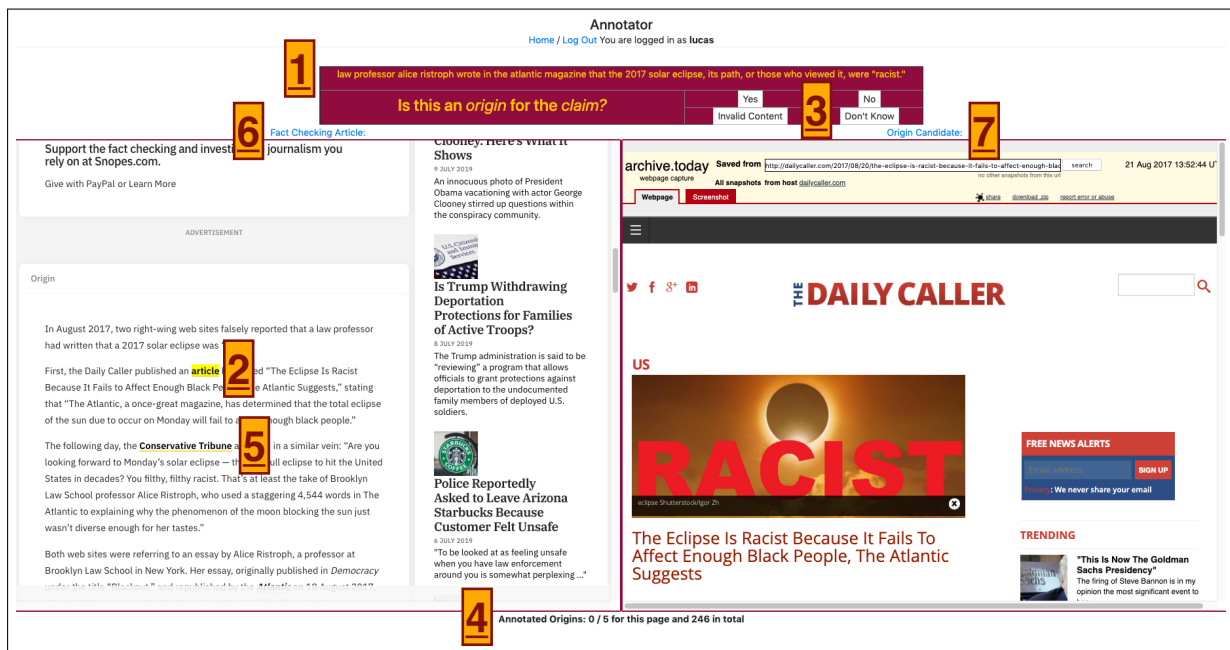


Figure 1: Annotator Interface

Figure 1 shows the Veritas Annotator as it is rendered by a web browser. Most of the screen space is used to display the FC article, on the left frame, and the origin candidate article, on the right. It also delivers information that may be important in order to ask the task question:

1. This section on the top of the annotator displays the Claim checked by the FC article. It is always visible, so there is no need for the user to search for it in the left frame.
2. The highlighted hyperlink in the FC article indicates which Source Page is being considered as the origin candidate at the current moment, this hyperlink's content is what is displayed on the right frame.
3. On the right-upper part of the screen, the user can find the four possible options for annotation, described separately on subsection below.
4. The counter of Annotations for the current user.
5. Other origin candidates hyperlinks for that same FC article. If clicked, the content of that link will be displayed on the right and from that point, the annotation will be regarding the newly selected origin candidate.
6. A hyperlink to the FC article.

7. A hyperlink to the origin candidate.

5.2 The Annotator Interface

On the first access of the Annotator¹³, users have to register with a unique username and password. Returning users should login with the same credentials they have registered before. This ensure no user will annotate the same OC more than once while also providing ways of evaluating the efficacy of the method by analyzing user's label allocation distributions and inter-user agreement.

Once logged in, and after every annotation is done, the user interface automatically requests from the Veritas Annotator a new origin candidate to be displayed and annotated, the selection of which entry should be assigned to each user has a randomness factor to it - to avoid any possible bias of storing order - but also follows a priority list: Initially, the Annotator ignores the OCs already annotated by that user, then it prioritizes the ones that were annotated twice, and amongst those, the ones that were given a "YES" by the other users. If there are no entries with two annotations, the priority goes to the ones with one annotation, and then to the ones with no annotations. After all the origin candidates were annotated three or more times, the annotator then retrieves the entry with the least number of annotations and displays it to the user.

¹³<http://veritas.annotator.insight-centre.org/>

The priority rules are defined this way so that a third annotator can break the tie for any OCs with two opposing annotations, to avoid having a single annotation of some OC (not a good idea as it means the validity of the annotation relies entirely on a single annotator), and to have as many annotated OC as possible (in that order). OCs that were annotated "yes" by other annotators also have a higher priority since that is our target class, in other words, identifying the origin means selecting the origin candidates that were labeled as "YES" by the majority of users that annotated it.

The original intention was for the web pages (both FC article and OC) to be retrieved upon request during the annotation process. It became evident that this approach would introduce a lot of idle time for the tool, which could make the task extremely tedious for annotators.

Initially, after selecting which OC should be displayed, the Annotator would then request and display the content of both the OC and the FC page URLs. That approach was generating many request and exhibition errors and more importantly, was increasing the time between annotations enormously, given that some OC pages are not hosted on their original addresses anymore but instead loaded from web archives. Since the list of FC and OC URLs that needed to be examined was known beforehand, a better approach would be to retrieve the webpages' HyperText Markup Language (HTML) code in advance and store them on the server, so that when requested by the user interface of the annotator they could be readily available. By performing this change, an overall decrease in the loading time was noticeable while also avoiding the need for the same site to be retrieved more than once, which accelerated the development, testing and evaluation.

On the upper part of the Annotator main screen there a table where the Claim analysed in the FC article is always visible and by the right side of this box, the four possible answers for the task question "Is this and origin for the claim?" are presented in the form of buttons. The instructions for when each button should be selected were extracted from the annotator guidelines¹⁴ and are presented below. Because of a space limitation, only one example is displayed in this article, although a variety can be found also within the an-

¹⁴veritas.annotator.insight-centre.org/guidelines

notator guidelines.

YES If the origin candidate article presented in the right suits the definition of origin, the a "YES" should be selected.

Invalid Content The user should select this option in the unusual case in which the presented content is not readable, either due to a failure of the Annotator to make a request, encoding or language related problems.

NO When the origin candidate page is displayed correctly but the content of it does not fall into the definition of origin.

I Don't Know For the cases where the user is not sufficiently assured about what is being stated in either the claim or the OC page.

Right below the box containing the claim and the buttons, the bigger part of the screen is vertically split into two frames displaying the FC page and the OC side-by-side. Above each frame there is a hyperlink not only indicating which frame displays which article but also allowing the user to access the content of that page directly. On the very bottom of the page, a count informs the user of how many OCs they have annotated in relation of the total of OCs of the current FC page and in total.

The development of the annotator had its own issues. As some FC Agencies have been operating for more than a decade, it was only natural to expect different website layouts and variance in many aspects, such as the type of encoding used in the sites, usage of HTML tags, classes used for verdict, structure, etc. Also, since we have no previous information about the origin candidate websites, they can be from any domain. Consequently, the retrieval, storage, and then display of HTML code in the Annotator lead to various issues as invalid references to resources and overlay cookies acceptances messages, request redirection, etc. The code¹⁵ used to develop the tool is publicly available.

6 Results

Shortly after the end of the Annotator's development stage, a gathering was organized with volunteers from different backgrounds to collect annotations. In total, 10 people participated and 2222

¹⁵<https://github.com/lucas0/Annotator>

annotations were made, in regards to 459 unique FC articles and 943 unique origin candidates. The quality of the verification task is controlled by majority voting, when considering only origin candidates there were annotated at least 3 times, we restrict the number of entries to 546, from where only 108 had “yes” as the majority votes. This is the initial number of documents of the final version of our gold standard dataset. As more annotations are done, this number will increase. There were also 56 other origin candidates that received more “yes” votes than “no”, “invalid content” or “I don’t know”, but did not reach the minimum number of votes of 3, recommended by crowdsourcing studies (Hsueh et al., 2009).

The inter-user agreement score, computed using Fleiss’ Kappa¹⁶ (a multi-user version of Cohen’s Kappa¹⁷) yielded approximately 0.16 as result, demonstrating a slight agreement between annotators.

This is not a sufficiently large number so other annotation sessions and events will still be organized in order to obtain more gold standard entries, although improvements in the linguistic-based fake news classifier could be seen and initial development of the mentioned automatic origin identification model was made possible.

7 Conclusion and Future Work

In general, this article describes the struggles of creating the first-of-its-kind Veritas dataset, intended for the task of automatic Fake News detection, which was our initial point. It also describes how that dataset creation process led us to the creation of an Annotator Interface, with its particular difficulties.

By performing this work, we expect to contribute not only with a new valuable language resource, but also with the ongoing work of other researchers also creating their own datasets, by describing the variety of different approaches implemented and evaluated.

Besides the inclusion of pages from agencies other than Snopes, we can see little to none improvement to be done in the Annotator itself. A higher inter-user agreement is desired but hard to obtain, given the high subjectivity of the annotation task, although perhaps a reformulation of

¹⁶https://en.wikipedia.org/wiki/Fleiss%27_kappa

¹⁷https://en.wikipedia.org/wiki/Cohen%27s_kappa

the guidelines providing more defined instructions could lead to an improvement on the Fleiss’ Kappa score.

The results achieved so far are considerable, and the ramifications of them into future work, exciting. To start with a bootstrap process, in which a binary classifier is being trained on the manually labeled OCs from the Veritas Annotator in order to perform the origin identification task automatically. Depending on the “certainty” - how close the predictions are to 1 - of this classifier, an OC could be automatically labeled as the origin, or sent to the group of entries to be manually annotated, from where more training input is generated, increasing its accuracy. This is a closed loop where the time spent by the human annotator is minimized while the results are enhanced both in quantity and quality.

Another application of this dataset is the already mentioned fake news classifier based on linguistic features (Azevedo, 2018) those two works are already being implemented and the initial results are promising, but out of the scope of this publication.

Additional data enrichment can be done by mapping Veritas Attributes to the schema:ClaimReview¹⁸ tags as they are being used by other authors (X Wang and C Yu and S Baumgartner and F Korn, 2018) and solidifying as a convention.

References

- Bill Adair, Chengkai Li, Yang Jun, and Cong Yu. 2017. Progress Toward the Holy Grail: The Continued Quest to Automate Fact-Checking. *Computation + Journalism Symposium*, (September).
- John Arbuthnot and Jonathan Swift. 1874. *The Art of Political Lying*. K. Tompkins.
- Lucas Azevedo. 2018. [Truth or lie: Automatically fact checking news](#). pages 807–811.
- Mevan Babakar and Will Moy. 2016. The state of automated factchecking. *Full Fact*.
- Charles F Bond Jr and Bella M DePaulo. 2006. Accuracy of deception judgments. *Personality and social psychology Review*, 10(3):214–234.
- Leon Derczynski and Kalina Bontcheva. 2014. PHEME: Veracity in digital social networks. In *UMAP workshops*.
- William Ferreira and Andreas Vlachos. 2016. Emergent: a novel data-set for stance classification. In

¹⁸<https://schema.org/ClaimReview>

- Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: Human language technologies*, pages 1163–1168.
- JF George and BT Keane. 2006. Deception detection by third party observers. In *deception detection symposium, 39th annual Hawaii international conference on system sciences*.
- Jeffrey T Hancock, Jennifer Thom-Santelli, and Thompson Ritchie. 2004. Deception and design: The impact of communication technology on lying behavior. In *Proceedings of the SIGCHI*, pages 129–134. ACM.
- Benjamin D Horne and Sibel Adali. 2017. This just in: Fake news packs a lot in title, uses simpler, repetitive content in text body, more similar to satire than real news. In *Eleventh International AAAI Conference on Web and Social Media*.
- Benjamin D Horne, Sara Khedr, and Sibel Adali. 2018. Sampling the news producers: A large news and feature data set for the study of the complex media landscape. In *Twelfth International AAAI Conference on Web and Social Media*.
- Pei-Yun Hsueh, Prem Melville, and Vikas Sindhwani. 2009. Data quality from crowdsourcing: a study of annotation selection criteria. In *Proceedings of the NAACL HLT 2009 workshop on active learning for natural language processing*, pages 27–35. Association for Computational Linguistics.
- Anoop K, Manjary Gangan, Deepak P, and Lajish V L. 2019. *Leveraging Heterogeneous Data for Fake News Detection*, pages 229–264.
- Tanushree Mitra and Eric Gilbert. 2015. Credbank: A large-scale social media corpus with associated credibility annotations. In *Ninth International AAAI Conference on Web and Social Media*.
- Ndapandula Nakashole and Tom M Mitchell. 2014. Language-aware truth assessment of fact candidates. In *ACL (1)*, pages 1009–1019.
- Kashyap Popat, Subhabrata Mukherjee, Jannik Strötgen, and Gerhard Weikum. 2017. Where the truth lies: Explaining the credibility of emerging claims on the web and social media. In *Proceedings of the 26th International Conference on World Wide Web Companion*, pages 1003–1012. International World Wide Web Conferences Steering Committee.
- Kashyap Popat, Subhabrata Mukherjee, Andrew Yates, and Gerhard Weikum. 2018. Declare: Debunking fake news and false claims using evidence-aware deep learning. *arXiv preprint arXiv:1809.06416*.
- Martin Potthast, Johannes Kiesel, Kevin Reinartz, Janek Bevendorff, and Benno Stein. 2018. *A stylistometric inquiry into hyperpartisan and fake news*. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 231–240, Melbourne, Australia. Association for Computational Linguistics.
- Georg Rehm, Julian Moreno-Schneider, and Peter Bourgonje. 2018. Automatic and manual web annotations in an infrastructure to handle fake news and other online media phenomena. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*.
- Hany M SalahEldeen and Michael L Nelson. 2013. Carbon dating the web: estimating the age of web resources. In *Proceedings of the 22nd International Conference on World Wide Web*, pages 1075–1082. ACM.
- Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. 2017. Fake news detection on social media: A data mining perspective. *SIGKDD Explorations*, 19:22–36.
- Aldert Vrij. 2000. *Detecting lies and deceit: The psychology of lying and implications for professional practice*. Wiley.
- M. Mitchell Waldrop. 2017. News feature: The genuine problem of fake news. *Proceedings of the National Academy of Sciences of the United States of America*, 114 48:12631–12634.
- William Yang Wang. 2017. ”liar, liar pants on fire”: A new benchmark dataset for fake news detection. *arXiv preprint arXiv:1705.00648*.
- Xuezhi Wang, Cong Yu, Simon Baumgartner, and Flip Korn. 2018. Relevant document discovery for fact-checking articles. In *Companion Proceedings of the The Web Conference 2018*, pages 525–533. International World Wide Web Conferences Steering Committee.
- X Wang and C Yu and S Baumgartner and F Korn. 2018. Relevant document discovery for fact-checking articles. *Companion of the The Web Conference 2018 on The Web Conference 2018 - WWW 18*, page 525533.