Rogue Algorithms: Using AI to Track the Spread of Disinformation

Jimmy Mulder[1,*,†], Librecht Kuijvenhoven[1,†], Stan Meyberg[1], Stefan Leijnen[1].

[1]*Utrecht University of Applied Sciences, Heidelberglaan 15, 3584 CS, Utrecht, The Netherlands* [1]
ORCiD ID: Jimmy Mulder http://orcid/org/0000-0001-9681-863X
Stefan Leijnen https://orcid.org/0000-0002-4411-649X

Abstract
Disinformation has become a growing problem in the digital age, and the rise of generative AI will likely only increase its ubiquitousness. Human fact-checkers are able to qualitatively debunk a tiny fraction of fake news, but they cannot keep up with the vast amounts of disinformation that is unleashed every day. There is a demand for automated tools to aid the process of identifying (potential) disinformation. In this paper we suggest a new quantitative approach, using a recursive algorithm based on Large Language Models to provide insight into the spread of disinformation articles. Our program identified the original source for 200.000 articles spread across more than 7000 websites. This information can be used to assess the trustworthiness of websites that host news articles.

Keywords
Disinformation, Misinformation, Algorithms, LLM, Explainability

1. Introduction

In their Global Risks Report 2024, the World Economic Forum identified misinformation and disinformation "as the most severe global risk anticipated over the next two years", citing the disruption of electoral processes, growing distrust, and increasingly polarized views [1]. While the authors warn for the risks of inaction, they also note that "There is a risk of repression and erosion of rights as authorities seek to crack down on the proliferation of false information", highlighting the need for an efficient remedy which preserves the rights of individuals.

A popular method to curb the effects of disinformation is the employment of so called 'fact checkers' [2], who take an article or belief and deconstruct its arguments one by one, using reputable sources and expert opinions. This is a costly process in terms of manual labor, and with hundreds of new disinformation articles being published daily (see results) it is not feasible for fact checkers to keep up. Furthermore, fact checking articles only reach consumers who visit reliable websites in the first place, reducing their effectiveness [3].

Much is unknown about the role of malicious algorithms in the spread of disinformation. Researchers have found a significant influence of 'bots' on social media [4]; does this apply

1∗ Corresponding author.
† These authors contributed equally.
jimmy.mulder@hu.nl (J. Mulder); stefan.leijnen@hu.nl (S. Leijnen)
0000-0001-9681-863X (J. Mulder); 0000-0002-4411-649X (S. Leijnen)

to news websites as well? Are malicious programmers using algorithms to autonomously find and re-post disinformation? Anecdotal evidence such as 'authors' publishing more than a thousand articles per year suggests that these practices occur, but the scale is unknown. The European AI act [5] requires low-risk applications to be transparent about the algorithms they employ, and prohibits algorithms to pose as persons; identifying malpractices can provide an opportunity to legally curb the spread of disinformation while preserving the right to free speech.

To handle the large volume of disinformation, software solutions capable of autonomously detecting disinformation are increasingly being researched and deployed [6]. These algorithms can be divided roughly into two strategies: one focusses on the message, determining the truth value of an article by looking at its content; the other focusses on the messenger, by estimating the trustworthiness of the source.

While the first strategy seems fairer for not dismissing an article based on its publisher alone, it is much harder to guarantee fairness this way. This strategy requires using a Large Language Model to recognize patterns in disinformation articles, but these LLMs may have unknown biases [7]. Additionally, data scientists may add their personal biases to a model when selecting datasets to train a disinformation classifier [8]. After being trained, such a classifier will almost certainly be a black box, which makes it very difficult explain why a certain article was deemed disinformation or not. Also, if such a classifier is made public, a malicious disinformation creator may use it to train a model to fool the classifier, resulting in a GAN-like competition. And finally, even if 'debunking' can be automated, its effectiveness is still debated [9]. We believe that a responsible AI solution should be transparent, explainable and free of bias, which cannot be guaranteed with this method.

While the second strategy is less precise in its assessment of any single article, the algorithms used for this type of program allow for greater transparency and explainability, which in turn lowers the risk of (unknown) biases and increases objectivity [10]. As we will demonstrate, a user of our program is able to inspect how news websites interact with each other to gain a qualitative insight into their trustworthiness. This offers a more responsible tool to journalists, scientists and internet users who wish to identify potential misinformation.

2. Method

Our application creates a network graph based on the number of shared articles between websites. The application consists of three algorithms: one to determine the level of similarity between two articles, one to scrape websites for articles, and one to find possible duplicates of articles on other websites. These work together to form our 'backend' which produces data on the similarity and source of online articles; all data analysis (such as mapping the data into a visual graph) is done post-hoc in a different environment.

At the core of our application lies a Large Language Model (LLM), in our case based on the multilingual LLM MUSE by Meta [11], which can take two sentences as input and output their semantic similarity as a value between 0 and 1. In order to calculate the semantic similarity between two articles, we cross-compare every sentence in article A with every sentence in article B, and pair those sentences with the highest similarity score. We then

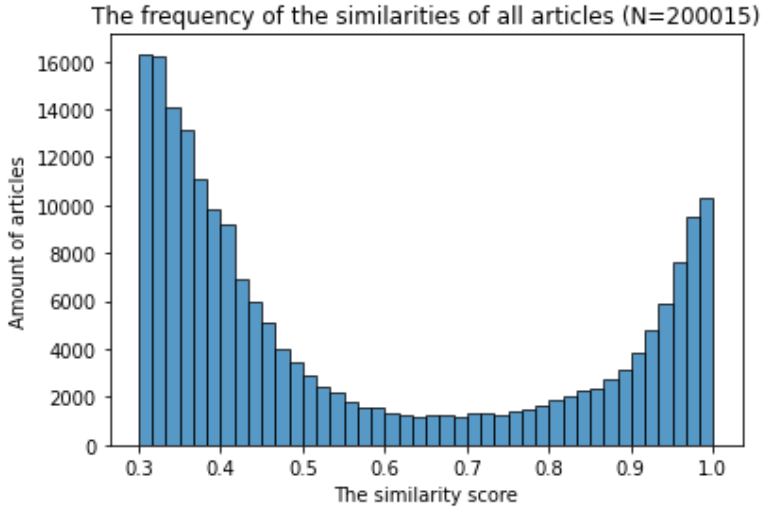
calculate the average similarity across all sentence pairs. The result is a similarity score between 0 and 1 for the entire article. Qualitative validation revealed that articles which are word-for-word the same receive a similarity score of 0.9-1.0, as expected. Articles that are very different score between 0 and 0.5.

The scraper algorithm produces a list of all the articles that a website has published. In most cases this list can be produced by looking at the sitemap of a website. In cases where no sitemap is found, a crawler is employed to crawl that specific website and gather as many links to articles as possible.

Our goal is to provide insight into the spread of disinformation and determine whether a website mostly produces original content or mostly reproduces content from other sources. Our third algorithm works by doing this analysis for one starting point (i.e. one website chosen by the authors) and then recursively adding more websites to the list.

Each recursive 'round' does the following:

- For every website on the list that has not been analyzed yet, do the following:
    - Use the scraper algorithm to find all articles that were published in the last two years.
    - For each of these articles:
        - Use a search engine (we used Bing Search) to find five potential duplicates of this article, based on the title.
        - Calculate the similarity score between the article and each potential duplicate, and use this score to determine whether these potential duplicates are really (semantically) close to identical. The threshold for two articles to be considered a match was set to a similarity score of 0.7, based on the distribution of similarity scores, as shown in figure 1.



In case two articles match, the website that hosted the newly discovered duplicate is added to the list. In future rounds, this website will then be analyzed in the same way.

Because the algorithm is recursive, it can theoretically run forever. Due to financial constraints we bounded the runtime of our algorithm to analyze 200.000 articles spread over more than 7000 websites. At this point there were roughly 70.000 websites and 7.4 million articles left on our 'to-do' list.

An interactive network graph was created from the database, showing the connections between all websites that have at least one article (with sufficient similarity) in common.

3. Results

Figure 2 shows a screenshot of our interactive visual graph which can be viewed at our website: https://uashogeschoolutrecht.github.io/RogueAlgorithmsVisialisation/. Here users can view each website as a node, with all the relevant articles attached. Users can also

Figure 1. A histogram of the similarity score distributions. Most potential duplicates found using bing search are either very similar or very different to the original, with only a

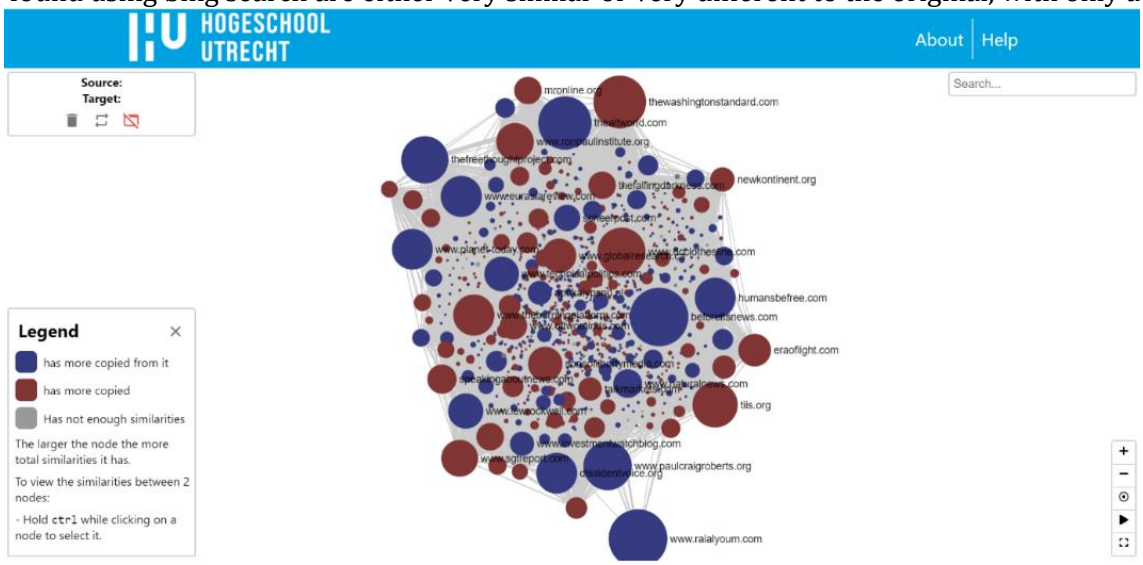


Figure 2. A screenshot of our interactive tool. Blue nodes host mostly original articles, red nodes host mostly duplicates. Nodes and links are clickable, which provides a list of articles.

click on a connection between two nodes to view the articles that they share, and can qualitatively inspect these articles to verify the decision of the algorithm. The data for this graph was acquired in December 2022, and articles may have been deleted since.

Quantitative analysis can be done in a separate environment. Since the runtime of our experiment was highly constrained for budgetary reasons, the resulting dataset of matching articles is a semi-random subset of all possible articles. As such, no conclusions can be drawn about specific actors, hubs or networks. However, some observations are of interest.

Most duplicates are shared within the first day after the original is published, usually within the first hour. However, there are outliers of up to three years. Another surprising observation is that although we started our analysis on Dutch websites, the algorithm quickly found translations in English, Arabic and other languages, creating a diverse dataset and showcasing the strengths of using a multilingual LLM. In one example, two Dutch articles (listing eight health benefits of apples) with different wording were both linked to an English article (which listed ten benefits), illustrating the models robustness.

4. Conclusion & Discussion

We have developed an algorithm that has proven to reliably and resiliently track the spread of (dis)information across websites on the internet. By employing a multi-lingual LLM we can detect translations, word replacements and other edits that would normally obfuscate the link between a source and its duplicates. We determined for every analyzed website how many of their published articles are original and how many are (mostly) duplicates of other – and which – websites.

This data could conceivably be used to calculate a kind of trustworthiness index; websites which copy many articles from untrustworthy sites can themselves be considered untrustworthy. Articles that are originally published by untrustworthy sources can be flagged as unreliable when they appear on other websites. These actions can inform internet users and legal representatives. However, this analysis fell outside the scope of our research.

The versatility of the used tools allows the application to be used in other ways. For example, authors and publishers can use this approach to detect plagiarism, although we did not compare our similarity scores to those given by conventional plagiarism detection software. Additionally, the LLM could be replaced with an image-based foundation model (such as Dall-E), which would allow the algorithm to track the spread of images across the internet. By applying similarity scores to the embeddings of images, such a tool would be resistant (to some extent) to any tampering with an image.

Within the context of disinformation, we believe that a quantitative analysis may also reveal qualitative links between websites (e.g. one author writing for multiple publishers). We have also done a rudimentary topic analysis: in our dataset, the words ‘corona’ and ‘vaccine’ occurred the most often in titles of articles, but ‘war’, ‘climate’ and ‘bitcoin’ were also popular. A more sophisticated analysis may reveal interesting patterns of thought within disinformation networks.

Some might consider it a downside that our algorithm completely ignores the content of articles in determining the trustworthiness of the source. In our view this is one of its major strengths: rather than reproducing the biases of a few programmers in an algorithm which determines what is ‘true’, we allow users to qualitatively assess any number of baselines based on their own expertise. This does mean that the quality of the initial baselines chosen by the user will have a significant impact on the long-term effectiveness of the algorithm as it progresses recursively. However, the transparency and explainability of our method ensure that any questions about the algorithm can be answered, providing a principled starting point in curbing the spread of disinformation.

Acknowledgements

We thank the Stichting Internet Domeinregistratie Nederland (Dutch Internet Domain registration Foundation) for providing the funds for this research.

References

[1] World Economic Forum, The Global Risks Report 2024. 2024. [Online]. Available: www.weforum.org

[2] C. Lim, "Checking how fact-checkers check," Research and Politics, vol. 5, no. 3, Jul. 2018, doi: 10.1177/2053168018786848.

[3] M. Hameleers and T. G. L. A. van der Meer, "Misinformation and Polarization in a High-Choice Media Environment: How Effective Are Political Fact-Checkers?," Communic Res, vol. 47, no. 2, pp. 227–250, Mar. 2020, doi: 10.1177/0093650218819671.

[4] M. Del Vicario et al., "The spreading of misinformation online," Proc Natl Acad Sci U S A, vol. 113, no. 3, pp. 554–559, Jan. 2016, doi: 10.1073/pnas.1517441113.

[5] "European Parliament P9_TA(2024)0138 Artificial Intelligence Act (COM(2021)0206-C9-0146/2021-2021/0106(COD)) (Ordinary legislative procedure: first reading)," 2019.

[6] P. Nakov et al., "Automated Fact-Checking for Assisting Human Fact-Checkers," Mar. 2021, [Online]. Available: http://arxiv.org/abs/2103.07769

[7] A. Abid, M. Farooqi, and J. Zou, "Persistent Anti-Muslim Bias in Large Language Models," Jan. 2021, [Online]. Available: http://arxiv.org/abs/2101.05783

[8] C. Winship and R. D. Mare, "Models for Sample Selection Bias," Annu Rev Sociol, vol. 18, no. 1, pp. 327–350, Aug. 1992, doi: 10.1146/annurev.so.18.080192.001551.

[9] M. pui S. Chan, C. R. Jones, K. Hall Jamieson, and D. Albarracín, "Debunking: A Meta-Analysis of the Psychological Efficacy of Messages Countering Misinformation," Psychol Sci, vol. 28, no. 11, pp. 1531–1546, Nov. 2017, doi: 10.1177/0956797617714579.

[10] European Commission, "High-Level Expert Group on Artificial Intelligence set up by the European Commission Ethics Guidelines for Trustworthy AI," 2018. [Online]. Available: https://digital-strategy.ec.europa.eu/en/library/ethics-guidelines-trustworthy-ai

[11] A. Conneau, G. Lample, M. Ranzato, L. Denoyer, and H. Jégou, "Word Translation Without Parallel Data," Oct. 2017, [Online]. Available: http://arxiv.org/abs/1710.04087