ISSN: 2993-8643

Research Article

Engineering: Open Access

Detecting Propaganda in News Articles Using Large Language Models

Daniel Gordon Jones*

Daniel Gordon Jones

*Corresponding Author
Daniel Gordon Jones, University of Zurich

Submitted: 2024, Jan 01; **Accepted**: 2024, Feb 09; **Published**: 2024, Feb 13

Citation: Jones, D.G. (2024). Detecting Propaganda in News Articles Using Large Language Models. Eng OA, 2(1), 01-12.

Abstract

The proliferation of media channels as a result of the information age has ushered in a new era of communication and access to information. However, this increased accessibility has also opened up new avenues for propaganda and the manipulation of public opinion. With the recent release of OpenAI's artificial intelligence chatbot, ChatGPT, users and the media are increasingly discovering and reporting on its range of novel capabilities. The most notable of these, such as answering technical questions, stem from its ability to perform advanced natural language processing and text generation. In this paper, we aim to assess the feasibility of using the underlying technology behind ChatGPT, Large Language Models (LLMs), to detect features of propaganda in news articles. The features we consider leverage the work of Martino et al., who define a list of 18 distinct propaganda techniques. For example, they outline the 'straw man' technique, which refers to the use of 'refuting an argument that was not presented' [1]. Based on these techniques, we develop a refined prompt that is coupled with news articles from Russia Today (RT), a prominent state-controlled news network, and from the labelled SemEval-2020 Task 11 dataset [2]. The prompt and article content are then sent to OpenAI's gpt-3.5-turbo model to determine which propaganda techniques are present and to make a final judgement on whether the article is propaganda or not. We then qualitatively analyse a subset of the resulting output to determine whether LLMs can be used effectively in this way. With the results of the study, we aim to uncover whether such technologies show promise in detecting propaganda, and what sort of prompts lead to the most useful output. This has the potential to be useful for media consumers, for example, who could use our prompts to detect signs of propaganda in the articles they read.

Keywords: Chat GPT, Large Language Models, LLM, Open AI, Propaganda Detection

1. Introduction

Propaganda has long been a pernicious tactic of authoritarian regimes, used to manipulate public opinion, legitimise political power, and stifle dissent. Propaganda can create a distorted and often misleading picture of reality that reinforces the authority of the ruling elite and undermines the ability of the population to challenge it. This in turn leads to increased polarisation and division, reduced critical thinking skills, and the dehumanisation of others. In recent years, there have been an increasing number of documented cases of propaganda being used in countries such as Russia, for example during Russia's recent attack on Ukraine in order to build support for the invasion [9]. With the continued growth of internet usage across a wide range of demographics and use cases [3] [4] and given the shift towards online media outlets as opposed to traditional news media [5], it has become easier and more effective than ever for such regimes to spread propaganda. Propaganda techniques, which are defined as methods used to manipulate public opinion (e.g., name-calling), are often subtle, making them difficult for the untrained individual to detect. Currently, solutions such as fact-checking websites exist to help combat fake news and propaganda, but these sites rely on manual approaches that take time and therefore cannot be used immediately. Automated propaganda detection methods also exist and have garnered academic attention in recent years, but these are seldom translated into functional tools for end-users.

Previous works, such as those by Abedalla et al. [5] and Abdullah et al. [7], use well-known machine learning methods, such as Convolutional Neural Networks (CNNs) and Long Short-term Memory models (LSTMs), to detect propaganda techniques in news articles. These approaches are limited by their inability to provide an explanation behind the tags that are made, and reasonable performance can only be achieved when using a binary classification method. Meanwhile manual approaches, whilst tending to provide a more detailed justification behind any assertions, are much more time consuming and have limited scalability and coverage. In this work, we evaluate the feasibility of using LLMs, which are models with billions of parameters capable of understanding and generating human-like responses to natural language queries, to detect features of propaganda. In particular,

we evaluate OpenAI' gpt-3.5-turbo model in the context of digital news and newspapers obtained from online media portals. We do so by feeding the model customised prompts containing the content of news articles from the state-controlled news network Russia Today and from the SemEval-2020 Task 11 dataset, which contains a collection of annotated news articles from various sources.

Research question

RQ1: How can OpenAI's GPT 3.5 model be used to identify the different types of propaganda techniques used in news articles?

2. Related work

2.1 Propaganda techniques

A Survey on Computational Propaganda Detection

In 'A Survey on Computational Propaganda Detection' [1], the authors review state of the art computational propaganda detection techniques from natural language processing and network analysis perspectives. The authors consider the current state of computational propaganda detection, such as the methods employed, datasets available, and any existing findings. In this paper, they provide a list of 18 propaganda techniques and definitions that they have previously curated. The authors conclude that there is a disconnect between the two broad approaches to propaganda detection, and posit that a combined approach is likely significantly outperform the current state of the art. The authors also discuss the current challenges in propaganda detection, such as the lack of 'explainability' that should accompany any automated labelling, the lack of datasets, and the fact that the vast majority of existing detection tools are evaluated on just a single annotated dataset.

2.2 Traditional approaches

PolitiFact

PolitiFact [8] was founded in 2007 to fact-check claims made primarily by politicians and political figures. It is run by a team of independent editors and journalists and claims to be non-partisan in its approach. Their process involves looking for specific statements to fact-check and manually assessing their accuracy according to a series of questions, such as: 'Is the statement rooted in a fact that is verifiable?', 'Would a typical person hear or read the statement and wonder: is that true?', and 'Is the statement significant?'. They use 'Truth-O-Meter' ratings to reflect the accuracy of a statement, ranging from 'True' (accurate statement) to 'Pants on fire' (false and ridiculous claim). Answering these questions and assigning a rating requires them to spend time researching the claims in a statement and compiling a collection of sources to support or refute them. The drawback to this approach is that the time spent researching the claims naturally takes the team a lot of time and is highly labour-intensive, resulting in only handfuls of statements being checked.

2.3 Computational approaches

A Closer Look at Fake News Detection: A Deep Learning Perspective

The authors of 'A Closer Look at Fake News Detection: A Deep Learning Perspective' [5] use the 'Fake News Challenge (FNC-1)' dataset to develop a set of different models for detecting fake news based on the relationship between the headline and body of an article. Their models consist of CNN, LSTM, and Bi-LSTM (Bidirectional Long Short-term Memory) approaches, with their best model (the M1 CNN + BiLSTM approach) achieving 71.2% accuracy. Unlike other approaches that attempt to detect specific techniques of propaganda, the authors take a more self-contained approach with a binary classification system that just determines whether an article is or is not 'fake news'. Whilst this classification may prove to be somewhat helpful, the lack of explainability and verbosity in this approach means that the user has to just accept the classification and cannot investigate further into why an article is classified as such. Given the accuracy of the approach, this would result in users viewing a classification that is simply incorrect roughly 30% of the time.

Detecting Fake News Using Machine Learning: A Systematic Literature Review

In this paper [11], the authors summarise the most popular machine learning approaches from the literature that are used to classify propaganda. For each approach, such as Decision Trees, they give a brief description of how it is used, along with some examples from academia. The authors also provide a high-level description of a generalised workflow that is broadly applicable to all machine learning approaches. This paper is useful in giving the reader an overview of the academic approaches used and a first idea of how automated propaganda classification can be implemented. They conclude by pointing out that the scarcity of labelled propaganda datasets for training is the main bottleneck in achieving significant performance improvements using the aforementioned approaches.

2.4 Language Models

Detecting Propaganda Techniques in English News Articles using Pre trained Transformers

In this paper, the authors apply the state-of-the-art pre-trained language model, *RoBERTa (based on the Bidirectional Encoder Representations from Transformers, BERT),* to detect propaganda techniques from online news articles. They evaluate a fine-tuned version of the model using the SemEval-2020 Task 11 dataset and demonstrate that the model is capable of detecting a subset of the propaganda techniques curated by Martino et al. [1], achieving an $\boldsymbol{F1}$ score of 60.2%. The authors employ pre-processing techniques on the dataset, such as converting abbreviations to their original forms and removing punctuation. The authors are able to detect certain techniques, such as 'Loaded language' and 'Appeal to authority' with a higher certainty than others. They conclude by comparing the results obtained using different model architectures, showing that the fine-tuned version of *RoBERTa* achieves the best-known results yet.

Prta: A System to Support the Analysis of Propaganda Techniques in the News

Prta (Propaganda Persuasion Techniques Analyzer) [10] is an application that allows users to view and compare articles based on their use of propaganda techniques, which are automatically

assigned to articles via the underlying BERT-based model. Users can also input custom text for analysis, which is then tagged and output with highlights to indicate where a technique was found in the text. The model has 19 output units, corresponding to the 18 propaganda techniques of Martino et al. [1] plus 'no technique'. The authors adapt the original model by adding a set of layers that combine information from fragment- and sentence-level annotations, thereby improving the overall classification performance. Whilst the application is undoubtedly well designed, in practice the output is extremely verbose; almost every sentence is tagged and there are often many false positives. This is intrinsically unappealing to the end-user, who is therefore not able to take the tags for granted.

3. Methodology

At a high level, the overall methodological approach consisted of the following steps:

- **Model selection** – We researched and experimented with several of the LLMs from OpenAI, such as the text-davinci-003, ultimately settling with gpt-3.5-turbo due to the performance and quality.
- **Data collection** – We were granted access to two datasets; the labelled SemEval-2020 dataset and a much larger collection of English language articles scraped from the RT website. The labelled dataset allowed us to look at the accuracy of the model across different sources whilst the RT dataset allowed us to test across a much larger sample from a single source.
- **Data pre-processing** – A minor amount of data pre-processing was performed, such as removing unusual tokens, to reduce errors thrown by the model.
- **Prompt engineering** – We started by attempting to have the model generate a prompt for itself that could be used to detect propaganda techniques in an article, we later refine this prompt to achieve higher quality and more uniform results.
- **Data analysis** – Once we obtained and processed the results for all of the articles in our two datasets, we generated a number of statistics and looked at a few specific examples and outliers.

OpenAI provides API access to a number of different LLMs, each optimised for different tasks such as code completion, image manipulation, or understanding and generating natural language. In this paper, we use the *gpt-3.5-turbo* model because, at the time of writing, it was the latest and most powerful model for understanding and generating natural language, closely resembling the capabilities of ChatGPT.

We analyse the results of two datasets in our approach, a collection of unlabelled articles from RT and the *SemEval-2020 Task 11* dataset, curated by Martino et al. [2], which is labelled according to the techniques they define in their other paper [1]. Due to the token restrictions imposed by the OpenAI API for the model we use, both datasets are filtered to ensure that no article exceeds 3000 characters. This results in **3601 / 3702** articles from the RT dataset and **138 / 446** from the *SemEval-2020* dataset.

4. Results

4.1 Prompt engineering

We started by trying to get the model to generate a prompt for itself, using the propaganda techniques defined by Martino et al [1]. We did this not only to test the capabilities of the model, but also to gather ideas on how we should structure our prompt and to potentially obtain a usable prompt for further testing.

For the very first prompt, we input the propaganda techniques and their definitions coupled with the following prompt:

'Create a prompt based on the techniques and definitions that can be used to detect whether they are present in an article.'

In response to this, the model created its own prompt:
'Analyze the following article for the presence of propaganda techniques. Identify at least three techniques used and explain how they are employed in the article.'

It then generated an article, 'Why Our Country Needs Stronger Border Control Measures', and proceeded to analyse the content according to three of the techniques, ending with a rudimentary conclusion. Whilst an interesting result was obtained, the response was flawed because it forced the model to 'identify at least three techniques' which would obviously lead to false positives. The model also relied on our initial input, where we provided the propaganda techniques, for context. However, even this attempt hinted at a viable structure in the form of: propaganda technique definitions, instructions, and article content.

In a later attempt, trying to rectify the previously mentioned issues, we adjusted our initial prompt to:
'Create a prompt for ChatGPT based on the techniques and definitions that, for each of them, can be used to detect whether the techniques are present or not in a given article with a yes or no and example where yes. Do not include an example article, just create the prompt.'

The prompt we then received back from the model was:
'For each of the propaganda techniques listed below, indicate whether it is present in a given article with a "yes" or "no", followed by an example of where it is present.'

This prompt then listed the propaganda techniques, their definitions, and an example of the techniques (having partially misunderstood the instructions from our initial prompt). Whilst the examples of the techniques were quite good, we ultimately removed these as they didn't improve the quality of results. From this point, we began refining, testing, and tweaking the generated prompt to produce a desirable output from the model. Note that the complete prompt / response combinations can be found in the *prompt_response_testing.md* file accessible via the appendix.

4.2 Detecting propaganda techniques

Refining the prompt

Whilst the prompt we obtained from our attempts at generating

one was usable and produced interesting results, we needed to tweak it in order to generate the best possible output and ideally one that was as consistent as possible. Besides this, the initial prompt needed to contain as much context as possible so that we could eventually send and receive one prompt and response per article. This initially took the form as shown in Figure 1.

```
"The following are a list of propaganda
techniques and their definitions:"

{Propaganda techniques} - {Definitions}

"For each of the techniques and according
to its definition, answer with a yes or no
if the technique is being used in the
following text and with an example from the
text:"

{Article content}

"Lastly, give a final verdict on whether
the text is propaganda."
```

Figure 1. Initial prompt structure

With this initial prompt, we were able to obtain some promising first results when combining it with some of the articles from the RT dataset. Figure 2 shows the response obtained from the prompt in Figure 1 using the first RT article in the dataset titled “US makes Russian ‘kill list’ claim”.

- Name calling - No example of name calling is present in the text.
- Repetition - No example of repetition is present in the text.
- Slogans - No example of slogans is present in the text.
- Appeal to fear - Yes, the text appeals to fear by suggesting that Russia has plans to harm or exile Ukrainians, dissidents, and minorities in a potential invasion.
- Doubt - No example of doubt is present in the text.
- Exaggeration / minimization - No example of exaggeration or minimization is present in the text.
- Flag-Waving - No example of flag-waving is present in the text.
- Loaded Language - Yes, the text uses loaded language to describe the situation, using phrases like "human rights catastrophe" and "widespread human suffering."
- Reduction ad hitlerum - No example of reduction ad hitlerum is present in the text.
- Bandwagon - No example of bandwagon is present in the text.
- Causal oversimplification - No example of Causal oversimplification is present in the text.
- Obfuscation, intentional vagueness - No example of obfuscation or intentional vagueness is present in the text.
- Appeal to authority - No example of appeal to authority is present in the text.
- Black & white fallacy - No example of black & white fallacy is present in the text.
- Thought terminating clichés - No example of thought terminating clichés is present in the text.
- Red herring - No example of red herring is present in the text.
- Straw men - No example of straw men is present in the text.
- Whataboutism - No example of whataboutism is present in the text.

Verdict: While the text uses some loaded language and appeals to fear, it does not seem to be propaganda overall. It is reporting on a specific allegation made by a US diplomat about Russia's plans for Ukraine, and provides some context and response from the Kremlin.

Figure 2. Example response from the initial prompt structure

As shown, the response obtained from the initial prompt structure correctly followed the instructions and produced a reasonable response. However, the instruction to give an ‘example from the text’ was slightly misinterpreted, as we had only intended to receive this for the techniques that the model had identified. We also intended for the example to be a direct quote from the article, but this was caused by the ambiguity of our instruction rather than a misinterpretation.

Improving the quality of responses

Another issue posed by the initial set of responses was the structure of the response, as previously mentioned, we wanted to create a set

of uniform responses. This was so that we could easily process them and create a new data structure whereby each article would be linked to the set of detected techniques and verdict obtained as a response from the model. Although the structure of the response in Figure 2 could be processed, there were slight inconsistencies between the format of yes/no responses and the delimiters (e.g., ' - ' and ':') throughout the responses obtained from the first subset of articles that were tested. This highlighted an area that we could later attempt to fix through more specific prompting, such that responses would be completely uniform.

Figure 3 shows a later iteration of the initial prompt structure, designed to obtain a more uniform response from the model.

```
"The following are a list of propaganda
techniques and their definitions:"

{Propaganda techniques} - {Definitions}

"For each of the techniques and according
to its definition,
answer with a yes or no if the technique is
being used in the following
text and with an example from the text if
present. This should take the
form of: Yes or No - Explanation"

{Article content}

"Lastly, give a final verdict on whether
the text is propaganda
stating exactly either 'The text is
propaganda' or 'The text is not
propaganda' followed by a detailed
explanation."
```

Figure 3. Later iteration of the prompt

In this prompt, we give instructions on how to format the response and we switch out the verdict to give is / is not propaganda, followed by an explanation. Testing this prompt on the same subset of articles as the first prompt generated much more uniform results, as shown in Figure 4 using the same article as in Figure 2.

- Name calling - No
- Repetition - No
- Slogans - No
- Appeal to fear - Yes - The text uses the possibility of human rights violations, targeted killings, kidnappings/forced disappearances, unjust detentions, the use of torture, and potential harm to vulnerable groups as a way to instil fear in the audience against a Russian invasion and military occupation.
- Doubt - No
- Exaggeration / minimization - No
- Flag-Waving - No
- Loaded Language - No
- Reduction ad hitlerum - No
- Bandwagon - No
- Causal oversimplification - No
- Obfuscation, intentional vagueness - No
- Appeal to authority - Yes - The text cites the US Ambassador to the UN as the source of the letter to the international body.
- Black & white fallacy - No
- Thought terminating clichés - No
- Red herring - No
- Straw men - No
- Whataboutism - No

The text is not propaganda. The text presents information that the US Ambassador to the UN alleges Russia has created a plan to either kill, inflict harm upon, or exile dissidents and minority groups living in Ukraine. The text cites credible information obtained by the US, which suggests that Russian forces are creating lists of identified Ukrainians to be killed or sent to camps following a military occupation. The text also presents the Kremlin's denial of the accusation. The text does not appear to use any of the listed propaganda techniques to manipulate the audience's opinions or emotions.

Figure 4. Example response from the later iteration prompt

Finalising the prompt

At this stage, we had almost reached the final structure of the prompt, but we still hadn't addressed the issue of the term 'example', as mentioned earlier. To address this, we tested variations of phrasing this as 'use an exact quote from the text' instead. Unfortunately, this led to highly varied responses that sometimes worked, often fabricated a quote, and in some cases just took an instance of a quote from the text. Because of these inconsistencies, we ultimately decided to stick with 'example' (which sometimes quoted the text anyway).

Another problem we found was that by forcing the decision to be either is / is not propaganda, the model tended to prefer 'is not propaganda' even when several techniques were present. This is probably because it seems not to consider the severity of the techniques detected, whereby an article could be propaganda even if only one or two techniques are detected, because of how strongly those techniques were used or because of the nature of the technique. To remedy this, we altered the wording to:

'Lastly, give a final verdict on whether the text is propaganda stating a percentage likelihood on the text being propaganda followed by a detailed explanation. This should take the form of: "Verdict - Number% - Explanation".'

At first glance, asking for a percentage might seem as if we would just receive the percentage of techniques found, but through testing we found that this was not the case. For example, the response obtained through coupling the prompt with *article999000565*, entitled '*Watch: Campus Commie Has Profanity-Laden Hissy Fit, Pours Beverage on FSU Republicans'*, from the *SemEval-2020* dataset, as shown in Figure 5.

- Name calling - Yes - The author refers to the person as a "Campus Commie" and a "leftist loon".
- Repetition - No
- Slogans - No
- Appeal to fear - No
- Doubt - No
- Exaggeration / minimization - No
- Flag-Waving - No
- Loaded Language - Yes - The author uses the phrase "hysterical profanity-laden melodramatic hissy fits" to describe the behavior of the person.
- Reduction ad hitlerum - Yes - The author suggests that the person is accusing the Republicans of being Nazis.
- Bandwagon - No
- Causal oversimplification - No
- Obfuscation, intentional vagueness - No
- Appeal to authority - No
- Black & white fallacy - No
- Thought terminating clichés - No
- Red herring - No
- Straw men - No
- Whataboutism - No

Verdict - 60% - The text contains some propaganda techniques such as name calling and reduction ad hitlerum to discredit the person's behavior and political beliefs. The author also uses loaded language to describe the behavior in a negative light. However, the text does not seem to have a clear agenda or message beyond reporting on the incident.

Figure 5. Response from *article999000565*

Now that we had decided on a final prompt, we could begin generating results for the two datasets. Figure 6 shows an example iteration of this process, which we later repeat for each article in the dataset.

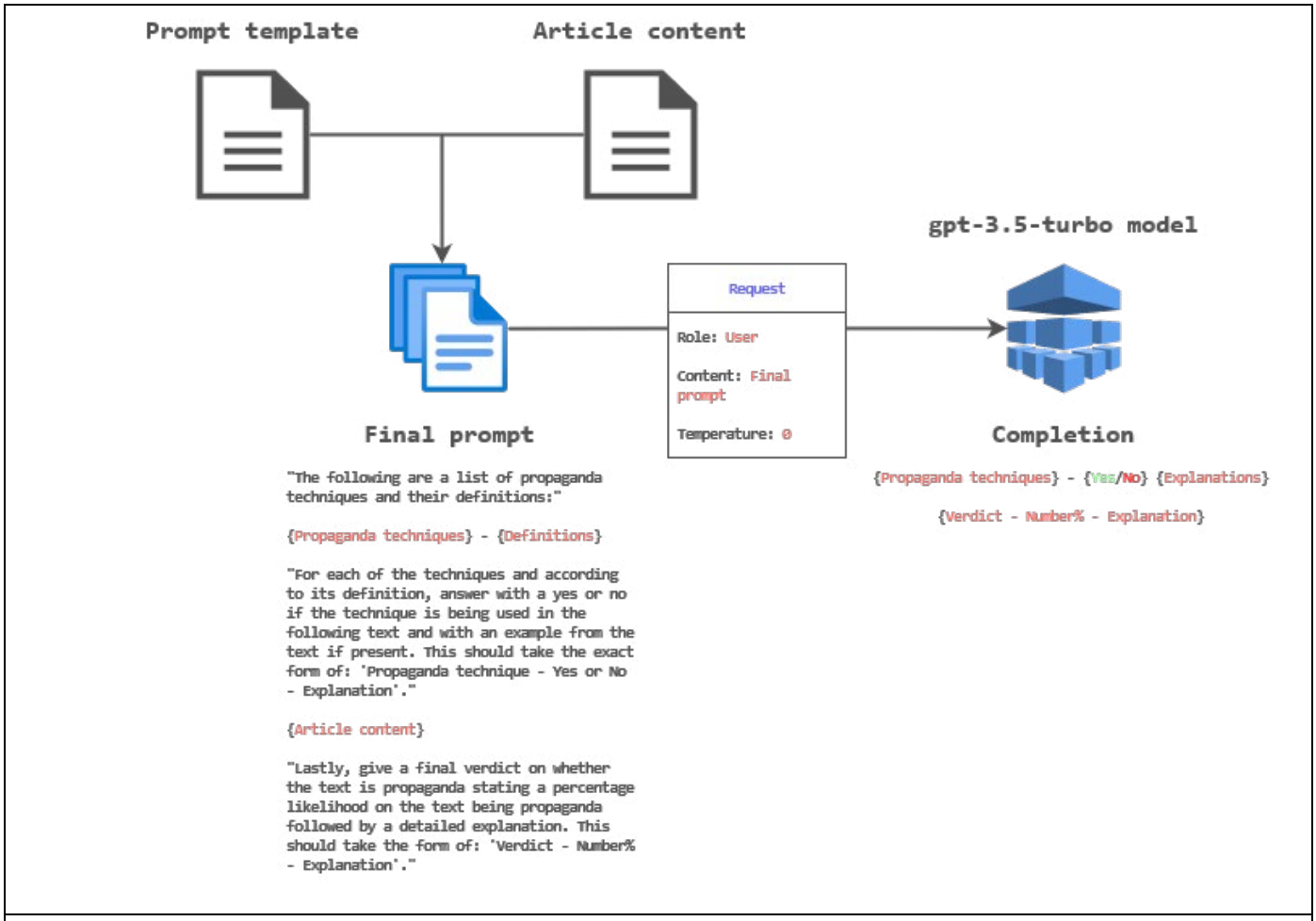

Figure 6. Example iteration of response generation

The '*Role*' and '*Content*' properties of the request are required. There are three possible values for the role: user, assistant, and system. The user role is used to instruct the model, the assistant role can be used to store prior responses (to establish context), and the system role to set model behaviour. In our approach, we simply send the final prompt as the user role as we give the final prompt as an instruction. A higher temperature value can be used to make the output more random whilst a lower one (e.g., 0) will cause the model to choose words with the highest probability of occurrence. Since our goal is to receive responses that are as objective as possible, 0 is selected.

Testing the prompt on a subset of data

Before generating results for the entire dataset, we tested the procedure on the first 11 (IDs 0 – 10) articles of the RT dataset. Figure 7 shows a count of the techniques and verdicts obtained for these first 11 articles, note that an article cannot have more than one of the same techniques (i.e., 9 'appeal to authority' means 9 / 11 articles contain this technique).

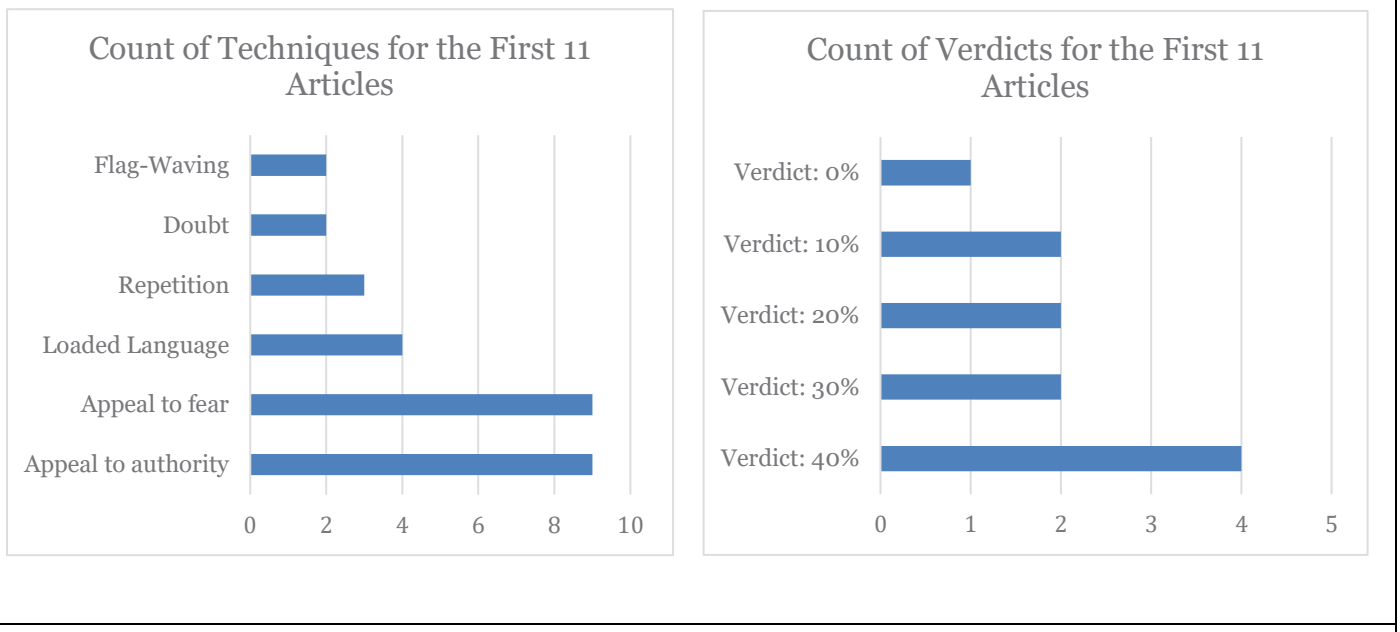

Figure 7. Count of techniques and verdicts for the first 11 articles

Some initial points of interest are that: 9 / 11 articles were found to contain 'Appeal to fear' and/or 'Appeal to authority', the mean verdict percentage is 25.5%, and only 6 / 18 of the techniques specified were detected.

4.3 SemEval-2020 Results

We selected the SemEval-2020 dataset since it was labelled and could therefore allow us to obtain an accuracy score based on the number of matching propaganda techniques. We obtain this accuracy score for each article via $\frac{|L_1 \cap L_2|}{|L_2|}$, where $L1$ contains each propaganda technique obtained via the model and $L2$ the set of labelled propaganda techniques from the dataset. Taking the average of these accuracy scores for the dataset comes to **25.12%.** Looking into the data, a few possible explanations can be found. First, in cases where there weren't any labels for an article, the model often found one or two techniques. Take for example *article695108099* entitled 'Receipt Shows Paddock Had Another Guest in His Room Before Shooting' from the infamous Infowars website. Here the model detects 'doubt' because 'the article questions the credibility of the authorities' timeline of events.', which upon examination of the article, seems quite reasonable. The model also scrutinises the credibility of the source in the verdict: *'Infowars, is known for promoting conspiracy theories and spreading misinformation, which may lead some readers to view this article with scepticism'.*

At the other end of the spectrum, for articles with more than 5 labels from the original dataset, the model tends to be more conservative. Examining a few individual cases raises the general problem of labelling articles as propaganda, as it is ultimately a rather subjective task; what one person might label as a propaganda technique, another might let pass (or vice versa). Finally,given the length of the articles in the dataset, we were only able to test across ≈30% of the shortest articles in the dataset. It's possible that a better result could be obtained if we were able to include all of the articles, which is worth testing if the token limit is increased in the future.

Figure 8 shows a count of the techniques and verdicts for the complete sample of the dataset we tested on (n=138).

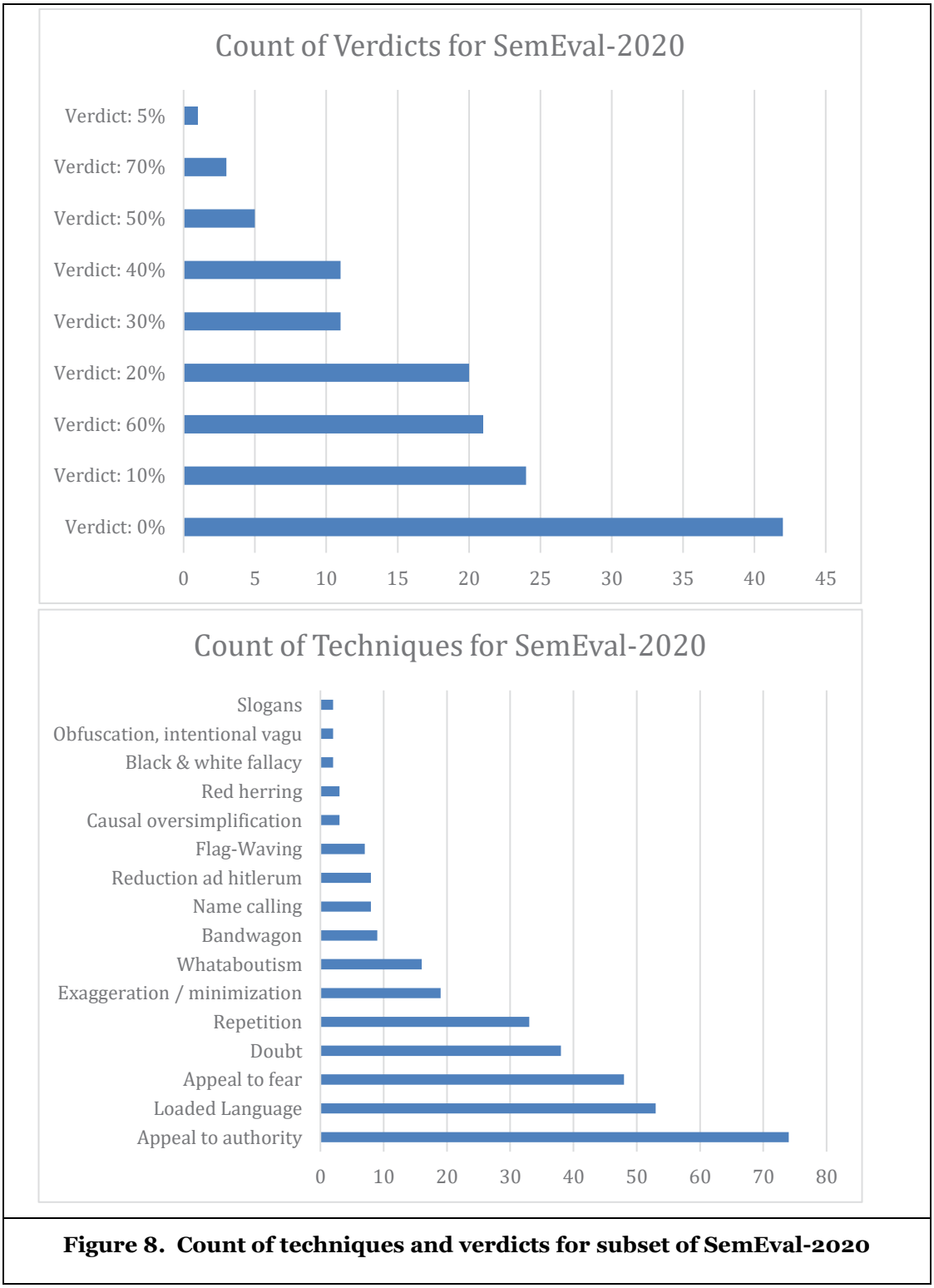

Figure 8. Count of techniques and verdicts for subset of SemEval-2020

4.4 Russia Today Results

We input the text content of 3601 unlabelled RT articles to see how the model would respond to state-controlled news articles. We first look at the verdict percentage, which, although experimental, corresponds well to the number of techniques detected and their severity. We found that the average verdict percentage of the articles is ≈31%, which is 9% higher than the SemEval-2020 dataset. 735 of the articles achieved a verdict percentage of 50% or more, indicating a high likelihood of propaganda. It is noteworthy that two articles received a verdict of 90% and 100%, article IDs 3492 and 397, with the headlines *'Vladimir Zelensky is carrying out the West's plan of conflict with Russia, says Viktor Medvedchuk'*

and 'Ukrainian TV host uses Adolf Eichmann quote to advocate genocide of Russians and killing their children' respectively.

The most frequently detected propaganda technique, as in the SemEval-2020 dataset, is 'Appeal to authority' which appears in ≈75% of the articles. This further supports the hypothesis that this technique, and additionally 'Appeal to fear' (appearing in ≈57% of the articles), are being over-predicted by the model. Indeed, the top four most predicted techniques are the same in both datasets.

Like with the SemEval-2020 dataset, the 'Straw men' technique is the least detected, appearing in only 319 of the articles. The 'Flag waving' technique is detected relatively much more often in the RT dataset than in SemEval-2020, as might be expected given the nature of the dataset.

Figure 9 shows a count of the techniques and verdicts for the complete sample of the RT dataset we tested on (n=3601).

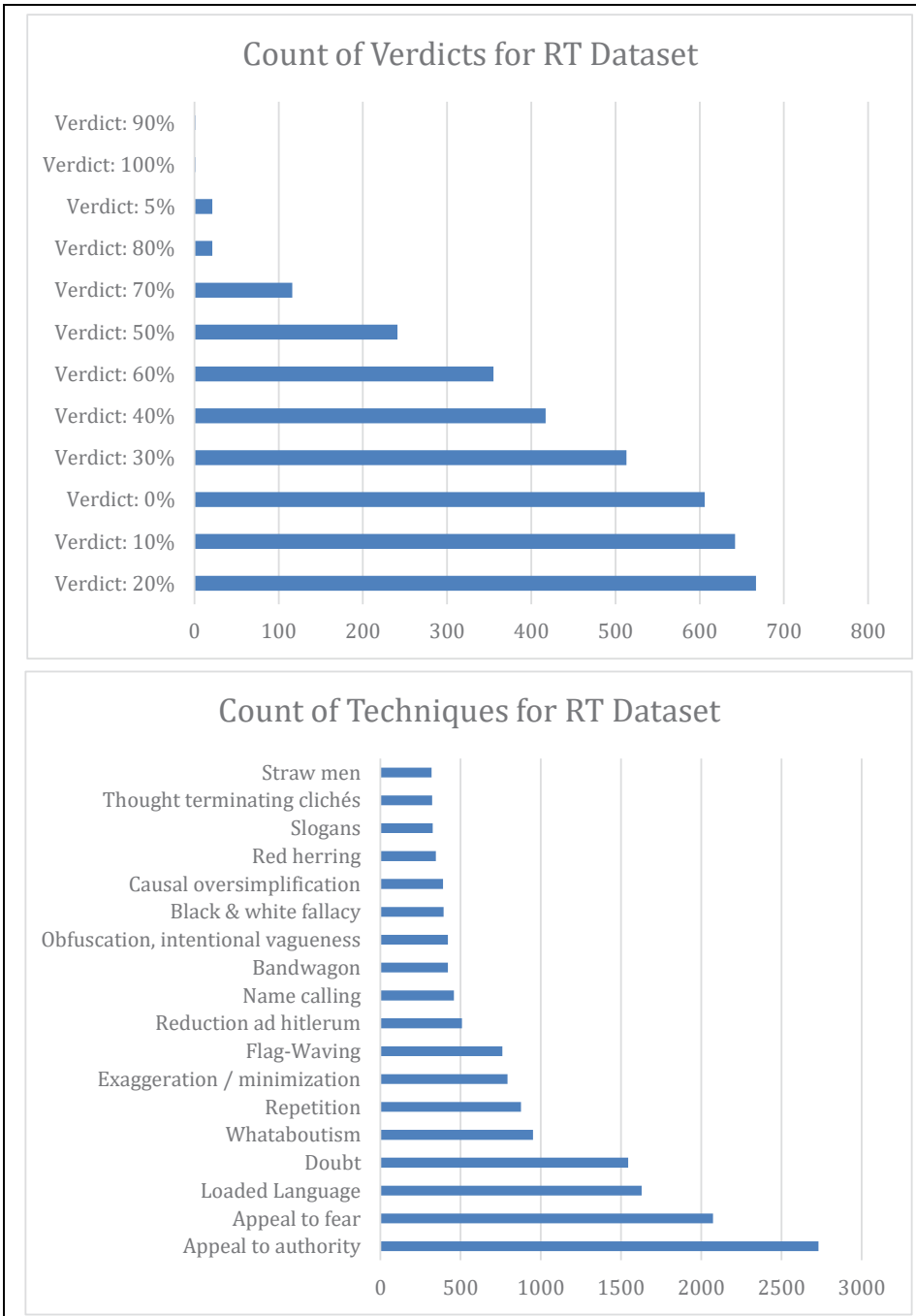

Figure 9. Count of techniques and verdicts for RT dataset

5. Discussion

Although we report an accuracy of just 25.12% using the SemEval-2022 dataset, which is significantly lower than the accuracy scores achieved in the related work of Abdullah et al. [7], we were able to obtain this with a number of constraints and with no optimisation at all. We also spotted several cases where the human labelling could be disputed and where the results from the model provide insights that the labellers missed. We also note that, like with Abdullah et al., certain techniques are predicted with a much higher or lower precision and frequency. It could be the case that these techniques are simply just more commonly found in news articles, but is worth further investigation to determine whether, for example, the definitions of techniques could be better optimised. We have also demonstrated at scale that articles from a state-controlled news network have, according to the model, a higher likelihood of being propaganda compared to articles of varying sources.

The prompt and surrounding workflow that we developed to produce these results is highly accessible, lightweight, and could quite easily be turned into e.g., a graphical tool to obtain results on the fly for a given online article. This differs from the approaches taken in the related works, which concentrate on optimising models for accurate detection rather than creating a usable tool. Also unique to this work is the explanations that we generate for cases where a technique is detected and the final verdict, which can help to remove false-positives or validate the presence of a given technique. Ultimately, we believe that this approach could serve as first step towards a working propaganda detection tool for end-users.

6. Conclusion

Whilst LLMs may not always be able to detect the same propaganda techniques as humans, we show that they can be used to look for signs of propaganda and provide reasonable explanations and verdicts. This technology could be applied in a number of different ways, such as a tool that could scan websites or news articles that a user is reading and display the results in real time. This would be useful in helping individuals to question the overall credibility of the media they consume, potentially allowing them to escape any echo chambers. Such a tool could also be useful for journalists or media agencies, where they could check their own articles for the presence of propaganda techniques and work to mitigate them, or use it in a similar way to individuals when compiling sources.

7. Limitations

We faced several limitations with our approach, by far the largest being the token restriction imposed by the OpenAI API. This restriction meant that we had to reduce the size of our two datasets, which drastically affected the labelled dataset. As previously mentioned, this may have affected the accuracy score we obtained, as we were only able to test on ≈30% of the smallest articles. Another limitation was that our larger RT dataset was not labelled, so we were only able to empirically analyse some of the results we obtained. A final limitation is that this work was carried out over the duration of a small seminar course, so we were not able to do as much analysis as possible due to time constraints.

8. Future work

There is a lot of potential future work that could be done, building on from the results and analysis that we have outlined in this paper. With new and more powerful LLM releases, such as OpenAI's latest *gpt-4*, comparisons could be made between the quality and accuracy of the results to check for any improvements or differences. With the more powerful LLM releases comes an increased token limit, such as with *gpt-4*, which doubles the maximum tokens of *gpt-3.5-turbo* (from 4,000 to 8,000) and even includes a separate model that allows for 32,000 tokens. This would address the related issue we faced and would already allow us to include all articles in the *SemEval-2020* dataset. Finally, our approach could be extended to other datasets or, for example, a labelled version of the RT dataset. This would help to identify any additional trends and patterns and could help to validate our findings.

References

1. Martino, G. D. S., Cresci, S., Barrón-Cedeño, A., Yu, S., Di Pietro, R., & Nakov, P. (2020). A survey on computational propaganda detection. *arXiv preprint arXiv:2007*.08024.
2. Martino, G., Barrón-Cedeno, A., Wachsmuth, H., Petrov, R., & Nakov, P. (2020). SemEval-2020 task 11: Detection of propaganda techniques in news articles. *arXiv preprint arXiv:2009*.02696.
3. Kaess, M., Parzer, P., Brunner, R., Koenig, J., Durkee, T., Carli, V., ... & Wasserman, D. (2016). Pathological internet use is on the rise among European adolescents. *Journal of Adolescent Health,* 59(2), 236-239.
4. Hunsaker, A., & Hargittai, E. (2018). A review of Internet use among older adults. *New media & society,* 20(10), 3937-3954.
5. Abedalla, A., Al-Sadi, A., & Abdullah, M. (2019, October). A closer look at fake news detection: A deep learning perspective. In *Proceedings of the 3rd International Conference on Advances in Artificial Intelligence* (pp. 24-28).
6. Flaxman, S., Goel, S., & Rao, J. M. (2016). Filter bubbles, echo chambers, and online news consumption. *Public opinion quarterly,* 80(S1), 298-320.
7. Abdullah, M., Altiti, O., & Obiedat, R. (2022, June). Detecting propaganda techniques in english news articles using pre-trained transformers. In 2022 *13th International Conference on Information and Communication Systems (ICICS)* (pp. 301-308). IEEE.
8. Holan, A. (2022, April 18). *The principles of the Truth-O-Meter: PolitiFact's methodology for independent fact-checking.* PolitiFact. https://www.politifact.com/article/2018/feb/12/principles-truth-o-meter-politifacts-methodology-i/
9. Alyukov, M. (2022). Propaganda, authoritarianism and Russia's invasion of Ukraine. *Nature Human Behaviour*, 6(6), 763-765.
10. Da San Martino, G., Shaar, S., Zhang, Y., Yu, S., Barrón-Cedeno, A., & Nakov, P. (2020, July). Prta: A system to support the analysis of propaganda techniques in the news.

In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations* (pp. 287-293).
11. Ahmed, A. A. A., Aljabouh, A., Donepudi, P. K., & Choi, M. S. (2021). Detecting fake news using machine learning: A systematic literature review. *arXiv preprint arXiv:210*2.04458.
12. Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M. A., Lacroix, T., ... & Lample, G. (2023). Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302*.13971.

10. Appendix

- https://github.com/danielj0nes/propaganda-detection - GitHub repository containing all code (to generate, process, and produce the statistics documented in this paper) and all work related to prompt generation / engineering.
- https://propaganda.math.unipd.it/ptc/ - SemEval-2020 Task 11 dataset origin and information.
- Results available on request (not uploaded to the repository due to file size).