الجمهـوريـة الجـزائريـة الديـمقراطيـة الشـعبيـة
PEOPLE'S DEMOCRATIC REPUBLIC OF ALGERIA

وزارة التعـليـم العالـي و البحث العلـمي
MINISTRY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH

جامعـة عمّـار ثليـجـي بالأغـواط
AMAR TELIDJI UNIVERSITY OF LAGHOUAT

كـليـة العلـوم
FACULTY OF SCIENCES
DEPARTMENT OF COMPUTER SCIENCE

Master's Thesis

Field: Mathematics and Computer Science
Specialty: Computer Science
Option: Data Science and Artificial Intelligence

By: Hacini Yasser

TOPIC

Detection of large scale influence operations using machine learning techniques.

Defended publicly on June 21st, 2025 before a jury composed of:

Dr. Tahar Bendouma	M.C.A	President
Dr. Leila Benarous	M.C.A	Examiner
Dr. Youssra Cheriguene	M.C.B	Supervisor

Thesis No. ..... – Academic Year 2024/2025

ملخص

مع زيادة الاعتماد على وسائل التواصل الاجتماعي كمصدر اخبار والانتشار الواسع لحروب الجيل الخامس التي تعتمد بشكل كبير على الدعاية والهندسة الاجتماعية والقرصنة وغيرها, نحن نشهد تزايدا كبيرا في عمليات الدعاية و نشر المعلومات المضللة التي أصبحت أسهل بفضل التقدم الكبير في تقنيات الذكاء الاصطناعي. وقد أدى ذلك إلى زيادة كبيرة في نسبة التعصب بين المجتمعات مما أسفر عن انخفاض التسامح بينهم. في هذه الأطروحة سوف نقترح تقنية تسمح لنا بتعداد التحيزات الموجودة في مجتمع واحد أو عدة مجتمعات باستخدام أسلوب يُعرف ب $(WEAT)$. نستخدم هذه التقنية بشكل أساسي لاكتشاف علاقات بين مختلف المواضيع التي تنشأ داخل هذه المجتمعات وكيفية تغير هذه الارتباطات مع مرور الوقت استجابة للتأثيرات الخارجية. تمكنت طريقتنا من إظهار ارتباطات واضحة داخل مجتمعات مختلفة، كما رصدنا تغيّرات مفاجئة في هذه الارتباطات تزامنت مع أحداث خارجية.

الكلمات المفتاحية: الذكاء الاصطناعي, الدعاية, حروب الجيل الخامس.

Abstract

With the ever-increasing popularity of social media, and the rising prevalence of fifth generation warfare that relies heavily on non-kinetic techniques such as propaganda, social engineering, hacking, etc. There has been a significant increase of large scale influence operation, which has become significantly easier to carry out with the recent advancements in generative AI and large language models, this has increased prejudice between communities, which has in turn decreased the tolerance seen between each other. In this dissertation we propose a technique which allows for the enumeration of both explicit and implicit biases found in one or multiple communities using a technique known as WEAT (Word embedding association tests), we mainly use it to find problematic associations made by these communities and how these associations change over time in response to outside influence. Using our technique, we were able to achieve an average P value of 0.033 and have been able to show clear problematic associations made by different communities, we also detected sudden shifts in associations which correlated with related outside events.

Keywords: Artificial intelligence, Propaganda, Word embedding, WEAT.

Outline

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Propaganda has been prevalent long before the spread of technology, there have been recorded instances of it dating back all the way to 515 BC in the Persian empire [1], Britannica defines it as "dissemination of information—facts, arguments, rumors, half-truths, or lies—to influence public opinion. It is often conveyed through mass media." [2], it is generally performed by governments on their citizens using various forms ranging from news, speeches, advertisements, etc.

In our context, influence operations are a more general form of propaganda, which does not limit the influencer to a state actor or a government, but to anyone with the ability to disseminate information to a group (or groups) of people, meaning that it includes individuals such as influencers, celebrities, politicians, or entities (such as companies or news channels). These actors can influence people using various means, for example a company can try to influence how consumers view it using PR (Public relation) campaigns, an influencer can use social media to spread certain views to their community, a news channel can create blind spots and cherry-pick news in order to shape a certain narrative for its viewers.

This is especially relevant during recent years owing to the rising level of polarization between many groups of people [3]. In addition to the increasing reliance on social media as a trusted source of news, which is especially liable to large scale influence operations due to how easy it is to reach large groups of people. In addition to the recent advances in generative AI, which make it nearly impossible to distinguish human made content from automated content [4], making us on the cusp of a new age where misinformation and manipulation is common place.

1.2 Context and background

Most recent techniques fall into one of two categories [5], the first being "text analysis", where researchers attempt to detect persuasion (or propaganda) from a piece of text using techniques such as classification of different propaganda techniques, this technique has potential to detect propaganda from a very small sample size (for example a single news article), however they suffer from very low accuracy and a risk of being potentially biased since the dataset used for training models are still labeled by people [6, 7]. The second technique, "network analysis", focuses on using metadata instead, for example: identifying small dense clusters of accounts –which can hint at it being a network of bots–, or trying to detect coordinated attacks –such as a group of accounts all posting about a single topic in a short period of time–, these techniques have been more successful at identifying large scale influence operations, for example botometer X [8] has been relatively successful at detecting bots during the time in which it was created.

However, there has been a lack of work attempting to merge these two techniques together, even though both techniques suffer from significant drawbacks. Network analysis techniques are capable of reliably detecting coordinated activity and clusters of bot accounts, however, this will no longer be the case as influence actors become aware of these techniques and adapt their influence operations in order to avoid detection, these techniques also lack the ability to identify the goal of the influence operation. Text analysis techniques have the potential to identify the goals and topics discussed in an influence operation, however they suffer from low accuracy, which is possibly caused by lack of context and/or implicit biases from the labeled datasets, They also suffer from the inability to generalize outside the context from which they were trained, for example, most models are trained on text data, making them unable to work with video, images, or audio data [5], they are also trained with the assumption that there exists a single ground truth that everyone must adhere to, making it not possible to use these models with other communities that have different beliefs or societal standards.

1.3 Research objectives

In this dissertation we aim to develop a novel technique for identifying how different communities perceive the world around them, how these different perceptions impact their relationship with other communities, and most importantly, the change in these perceptions over time allowing us to observe sudden shifts, which can, in turn, be attributed to having been influenced by something.

Our goal is to develop a technique that is able to aid previous techniques by overcoming some blind spots that they have, we focus primarily on providing a more understandable and objective approach to identifying the impacts of an influence operation on groups of people, as a result, our technique must meet the following criteria:

- It must be an unsupervised technique: in order to avoid implicit biases from labeled datasets, and to improve generalization between different communities which have different societal standards without the need for a large dataset encompassing all possible cases.
- It must clearly identify associations: for example if a community believes that A is heavily associated with B even though linguistically they don't have any association, our technique must be able to identify that association.
- It must accept different modalities of data: in order to allow us to capture any form of communication, not only text.

Furthermore, it is important to note that our technique does not try to identify the actor attempting to influence, nor does it attempt to be an early warning system, instead it aims to develop a technique that is capable of providing an objective metric to measure shifts in associations, meaning it can only detect when an influence operation has already occurred, and list how the perception of that community has changed over time.

1.4 Thesis organization

In the related works and methods chapter, we will explore a more in-depth overview of previous techniques and discuss their shortcomings, then we will lay down an abstract description of our problem which will help us identify a different approach to solving it, then, we will propose our technique for resolving it, and finally, we will give more detail on the datasets used, how data was processed, and how training and validation were done.

After that, in the results' chapter we will discuss the results we got from our technique, more specifically, we will measure the stability, statistical significance, interpretability, history, and the effects of the biases that we identified on downstream tasks such as sentiment analysis.

Finally, we will discuss the limitations of our technique, how we can resolve those limitations in future research, and the ethics regarding our dissertation.

Chapter 2

Background and related work

The term "Influence operation" encompasses a broad range of topics, including but not limited to: propaganda campaigns, PR campaigns, voting campaigns, etc. As a result, it is quite difficult to find research dedicated to a topic spanning such a wide range of definitions, which is why we have decided to focus on the topic of propaganda detection, because, not only does it align with our objectives, but the techniques used for detecting propaganda can easily be extended to work with the other topics.

2.1 Text based propaganda detectors

In this section we will explore a multitude of techniques who's main focus is to detect pieces of text that use different techniques to spread propaganda, there is no agreed upon list that presents all the propaganda techniques, however, most of these techniques use either logical fallacies or cognitive biases. Some examples of these techniques include:

- Appeal to emotion: A logical fallacy that is used to take advantage of a person's emotions in order to spread a certain message, for example attempting to arouse anger in a group of people in order to cause chaos, or showing images that promote sadness in order to sow hate against certain groups.
- Appeal to authority: A logical fallacy where information is assumed to be correct because the source of such information comes from a position of authority, this can be any form of authority, for example, a person may try to argue in favor of himself using the argument that he has been more successful, or a person may try to dismiss another person's arguments, using the argument that the person lacks knowledge, instead of attempting to confront the argument.
- The bandwagon effect: A cognitive bias where a person believes a piece of information because the majority of people around him believe in it, there are many variations of it, including herd mentality and groupthink. This is caused by a person's tendency to try and conform to the people around him.
- Confirmation bias: A cognitive bias where a person will tend to interpret patterns or search for information that confirms his beliefs, this is also related to another cognitive bias termed "cognitive dissonance" which is when a person suffers from significant mental load for having two contradictory beliefs, often resulting in them dropping the weakest one, even if it is more logical.

Some of these techniques are easy to detect, for example appeal to emotion and appeal to authority are relatively easy to recognize when confronted with a piece of text that uses them, they also sometimes use certain keywords and word combinations that make it easy for computational models to detect, for example, appeal to authority may have a structure similar to: "[position of authority] claims [piece of information]", and appeal to emotion may have a structure similar to: "[group] are responsible for [action that provokes an emotional response]".

However, other propaganda techniques may be significantly more difficult to detect without designing a system specific for that technique, a good example of this is attempting to detect misinformation, without access to live trusted information –used for fact checking claims–, it is nearly impossible to verify if a piece of text is spreading misinformation or not.

Even then, text based propaganda detectors have a clear advantage over other techniques, this advantage being, when using an accurate model, it is easy to get it to work on all platforms. This is because it only requires a piece of text no matter the platform, making it significantly easier to implement as, for example: a browser extension, or an open source addition to pre-existing pieces of software.

We present three different papers aimed at detecting propaganda or persuasion from pieces of text that we believe encompass most of the approaches taken by researchers.

Sprenkamp et Al. [6] used LLMs (Large Language Models) in order to classify different propaganda techniques on the SemEval-2020 Task 11 dataset [9] on the premise that LLMs are few shot learners making them easy to train on relatively small datasets.

The SemEval-2020 Task 11 dataset [9] is a collection of news articles spanning from mid 2017 to early 2019, these articles were gathered from 13 propaganda and 36 non-propaganda news outlets that were labeled by MB/FC (Media bias/Fact check) [10], table 2.1 provides an overview of the dataset size. From these news articles, two main tasks were proposed, the first being span detection, where models will try to identify the pieces of text that can contain one of the propaganda techniques used in the dataset, and the second task is identifying the propaganda technique used in the detected spans.

Table 2.1: Summary of the SemEval-2020 Task 11 dataset.

Partition	Articles	Average length in characters	Propaganda snippets
Training	371	5,681±5,425	6128
Development	75	4,700±2,904	1063
Testing	90	4,518±2,602	1790
Total	536	5,348±4,789	8981

Sprenkamp et Al used two LLMs, GPT-4 and GPT-3. GPT-3 was used in 3 different ways, first, the base GPT-3 model was tasked with identifying the propaganda techniques used in an article, secondly, another GPT-3 model was fine-tuned to output the labels of the propaganda techniques used in the article, and finally, the third model was prompted with no fine-tuning to use chain-of-thought and to reason about the choice of labels. The GPT-4 models were used in 2 ways, the first was using the base model with a prompt, and the second was using a prompt asking the model to use chain-of-thought. In both models, the approaches that use chain-of-thought and the base model were given one example of each propaganda technique.

They were able to achieve an F1 score of 58.11% using base GPT-4, compared to the –at the time– state-of-the-art model made by Abdullah et Al. [11] which achieved an F1 score of 63.40%, table 2.2 provides more details regarding the results found for every model.

Table 2.2: Results from the Sprenkamp et Al propaganda detectors.

Model	Precision	Recall	F1 Score
GPT-3 base	44.35%	44.00%	44.18%
GPT-3 CoT	48.62%	28.16%	35.66%
GPT-3 Fine-tuned	47.54%	32.48%	38.59%
GPT-4 base	52.86%	64.52%	58.11%
GPT-4 CoT	56.86%	57.82%	57.34%

Chernyavskiy et Al. [12] used the more recent SemEval-2023 Task 3 dataset [13].

This dataset –unlike the SemEval-2020 Task 3 dataset– is a multilingual dataset, which focuses on the detection of persuasion instead of propaganda, it is composed of articles in 9 languages (English, French, German, Georgian, Greek, Italian, Polish, Russian, and Spanish), collected from both mainstream media and alternative media using news aggregators (Google News and Europe media monitor [14]), these news articles were then annotated at both the text and document level with the help of 40 annotators, which were composed of media analysts, disinformation specialists, and NLP experts.

Table 2.3: Results from the Chernyavskiy et Al russian persuasion detector.

Model	Micro F1	Macro F1
XLM-RoBERTa base	24.11%	18.14%
XLM-RoBERTa discourse	31.20%	20.64%
Hromadka et Al [15].	38.68%	18.88%
Wu et Al [16].	31.84%	20.52%

They employed a modified transformer architecture that is discourse aware by relying on the Rhetorical Structure Theory (RST) [18] (shown in figure 2.1). They based their model on DeBERTa-v2 [19] for English and RoBERTa [20] for Russian. The competition utilized micro and macro F1 score, where macro F1 score corresponds with the performance of infrequent classes, their Russian model was able to achieve a macro F1 score of 20.64% and a micro F1 score of 31.20%, table 2.3 and 2.4 provides a more detailed summary of the results.

Barrón-Cedeño et Al. [21] Developed Proppy which attempts to classify news articles into five groups depending on their propaganda score, as a demonstration of their model they opted to use live news

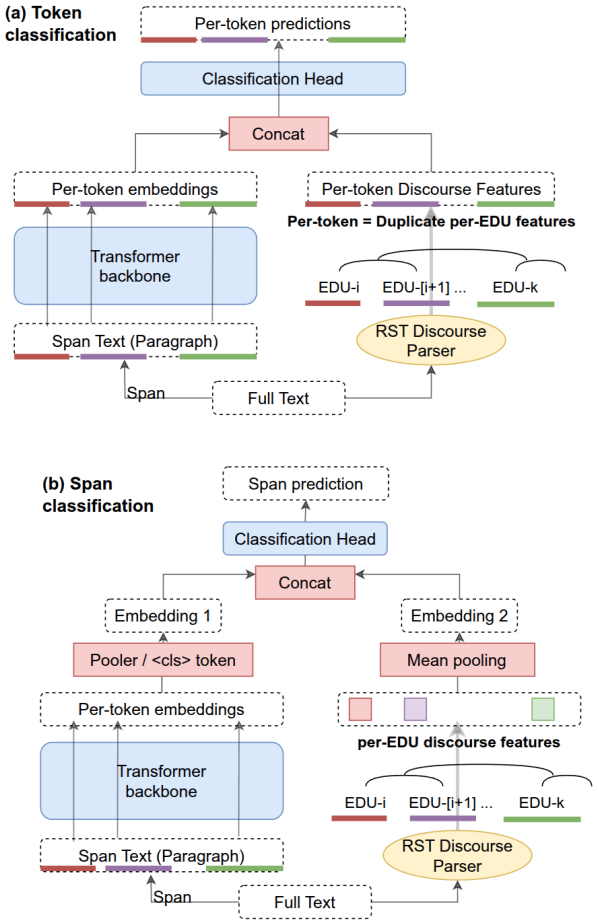

Figure 2.1: The modified transformer architecture used by Chernyavskiy et Al (Figure was taken from the cited paper).

Table 2.4: Results from the Chernyavskiy et Al english persuasion detector.

Model	Micro F1	Macro F1
DeBERTa base	32.87%	16.21%
DeBERTa NucSat	34.23%	16.92%
DeBERTa relations	37.51%	18.01%
DeBERTa position	36.20%	17.33%
Purificato et Al [17].	37.56%	12.92%
Wu et Al [16].	36.80%	17.19%

articles, where they use Qlusty [22] to represent the news article, then clustered them using DBSCAN [23], after that they discard near duplicates by calculating the jaccard coefficient [24] and keeping those that are below a certain threshold, the clustered posts are then passed to a classifier that will classify them into 5 bins depending on their propaganda level. The F1 score was then calculating by using propagandistic and non-propagandistic sources, by using five propagandistic sources and only character n-grams they were able to achieve an F1 score of 64.45% (Note: the paper used multiple trials with multiple results, some of which are better and some are worse, there were no criteria for selecting the mentioned F1 score).

Table 2.5 provides a comparison of the techniques mentioned above.

Even though Text analysis techniques have the potential to be highly useful, they suffer from relatively low scores making it not possible for them to be widely adopted for detecting propaganda, a potential explanation for the difficulty of this task is the fact that (even with LLMs which are able to store information) the model does not have enough context to recognize if a piece of text is propaganda or not. Another possible explanation for the low performance is that we could be seeing the effects of biases caused by having a labeled dataset, even though labelling is done by experts, it is possible that some implicit biases may have been incorporated to the dataset. Additionally, it is unknown if these models will perform as well when transferred to another medium, since different populations have different societal standards, and different languages have different structures, the models may have significantly more false positives and false negatives. As such it is important to address these issues by ensuring that the technique developed in this dissertation does not use a single piece of ground truth, and it must be

capable of addressing the multiple, potentially conflicting beliefs of communities.

Table 2.5: Comparison of different text based propaganda detectors.

Author	Technique	Dataset	Results
Barrón-Cedeño et Al.	unsupervised classification	Custom	64.45%
Abdullah et Al.	RoBERTa LLM	SemEval-2020	60.20%
Sprenkamp et Al.	LLM Few shot detection	SemEval-2020	58.11%
Chernyavskiy et Al.	Modified transformer	SemEval-2023	31.20%

2.2 Network based propaganda detectors

Network based techniques generally focus on detecting social media accounts spreading propaganda, as opposed to the propaganda content itself, and unlike text based techniques, research done on detecting these accounts have been more successful and were able to achieve higher scores. However, since there are no datasets available for this task, most techniques rely on unsupervised or semi-supervised techniques.

Davis et Al. [25] Developed the system BotOrNot which uses metadata from Twitter accounts in order to give a score of how likely it is that a certain Twitter account is a bot account, it uses many pieces of information, including: "Network" features such as retweets and hashtags, "User" features such as location and language, "Friends" features that include followers and following, "Temporal" features such as timing patterns, "Content" features such as linguistic cues, and "Sentimental" features that use a twitter specific sentiment analysis algorithms. These features are then trained using random forest on seven different classifiers on a list of known social bots identified by Cavarlee et Al. [26], which yielded a performance of 0.95 AUC (Area Under Curve).

Feng et Al. [27] used various LLMs with the TwiBot-20 [28] and TwiBot-22 [29] Datasets. They used multiple techniques, which include: Metadata based, Text based, Structure based, In-context learning, and Instruction tuning in order to make use of LLMs as bot detectors. They also explored misuse of LLMs and how they can be prompted to control an account in order to make it appear more human using techniques like refining the posts made by the bot account, adding or removing followers, selectively combining different sources of information, etc. Furthermore, they were able to achieve the best results for bot detectors by using majority vote ensemble using all the techniques mentioned previously, which achieved an accuracy of 89%, F1 score of 91%, Precision of 86%, and Recall of 97%. And even after using those LLMs in order to avoid detection, the ensemble methods did not suffer as much, where the Accuracy only dropped to 85% and F1 score to 86%, table 2.6 provides the results found before and after manipulation for the ChatGPT Detector and BotPercent [30].

Network based approaches show more promising results than text based approaches, since they are more capable at detecting bots, making them a more promising technique at identifying the actors responsible for performing a certain influence operation. However, they still suffer from a few drawbacks, the most obvious one being that they are not capable of recognizing the goal of the influence operation performed by bots, it is also very difficult to recognize what topics a botnet is attempting to influence since they tend to post about a wide range of topics different from those that they are targeting.

And even though these techniques have been more successful than text based techniques, that does not mean that they will remain as such, since most of them rely on the behavior and posts of the bot accounts. Past bots have been easily detected by both people and computational techniques, however this is changing quickly as these bot accounts are able to appear more human by creating more natural posts with the help of LLMs [31] and –potentially– by using modern evasion techniques.

Table 2.6: F1 scores of different detectors on the base accounts and accounts that were manipulated by ChatGPT and LLama.

Model	BotPercent F1	Emsemble LLama2-70B F1	Ensemble ChatGPT F1
Base posts	86.5%	65.9%	91.5%
ChatGPT manipulated	64.9%	NA	91.0%
LLama-70B manipulated	65.5%	NA	86.9%

Chapter 3

Contribution

3.1 Problem description

Bearing in mind the shortcomings of the techniques mentioned before, we will try to define a technique which aims to overcome these issues, this is done using an unsupervised approach that uses both network and text analysis techniques, it is also capable of seamlessly working with different communities no matter how different they might be. This is done by extracting the associations that these communities make between different concepts and analyzing them in order to identify any problematic associations or large shifts in beliefs. In order to do that, we first need to set up the necessary foundation to develop this technique.

3.1.1 High level overview

Before we can define how a person can be influenced, we first need to provide a definition of how information flows. Which can be described in an abstract way using the field of cybernetics, more specifically memetics and semiotics. Even though memetics is a controversial field because of its vague and abstract definitions (which are not of much use to the scientific community), its idea of how units of information (called memes) evolve through time and spread is useful for us –since our work will revolve around studying how points in an abstract space change over time–.

In summary, a meme spreads depending on how "contagious" it is, where the term "contagious" defines an abstract measure of how likely it is for that meme to be accepted by another entity, an entity in this context is also an abstract term describing things that take in a meme and can in turn output it. A meme is also capable of mutating over time potentially creating more contagious variants of itself, these mutations slightly change the information conveyed such that it is not noticeable as it spreads from one entity to another, but after many iterations it can cause significant and clear changes [32].

Semiotics on the other hand describe the manner in which memes are spread, where signs are either intentionally or unintentionally associated with one or multiple memes, for example, in everyday life, information is spread using language, and more specifically speech, meanwhile in different settings, such as art or music, we are able to convey information using different mediums such as audio or visual mediums. An example of this is a person feeling a certain emotion, they can either convey it intentionally by saying something like "I am happy", or unintentionally by a facial expression or an action, in this case the meme is the emotion that the person is feeling, and the sign is the speech, facial expression, or the action.

These signs are associated with certain memes using a mental model that is unique for each person, this mental model allows a person to associate signs with memes, or memes with other memes. It is also mutable by other signs that a person observes, either through conscious or unconscious cognition. However, these mutations are not easily predictable, since a sign could weaken an association for one person, strengthen it for another person, or have no impact. Making it incredibly difficult to predict what a person's reaction to a certain sign could be. But, this is made easier when attempting to predict the behavior of large groups of people, where groups of people tend to behave in a more predictable way, this is because people will attempt to conform with the rest of the group in different phenomenons such as "herd mentality" or "groupthink" [33, 34, 35].

3.1.2 Definition of an influence operation

From a high level, an influence operation is composed of 4 main parts (shown in Figure 3.1):

- The influencer: which is the entity performing the influence operation, they usually attempt to change the beliefs of a group of people in order to benefit themselves.
- The influence agents: which are the actors performing the influence operation, for example in the context of social media they can be bot accounts.

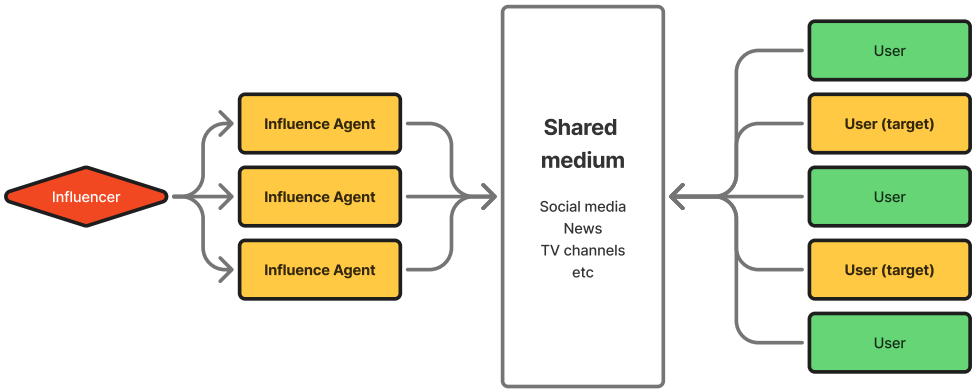

Figure 3.1: Simplified high level overview of an influence operation.

- The influence medium: which is where the influence operation is taking place, this can be by text, video, audio, or any other form of communication that is able to reach a wide range of targets
- The influence targets: they are the targets of an influencer, who's beliefs are being changed by an influencer, they are usually spread between other people that are not targets, the influence agents will attempt to reach all targets while preferably avoiding the rest.

The influence agents attempt to associate specific signs with each other which usually have either no association or an undesirable association. For example: we have 4 signs, sign A is positive, sign Z is negative, and signs I and J are considered linguistically neutral. The influencer wants to associate sign I with something negative, the influence targets have a pre-existing association of I with A and J with Z, the influence agents may attempt to convince the targets that there is an association between I and J, while making the association between I and A weaker, this makes it possible to remove the association between I and A, and create an association with I and Z, making the influence operation successful. Figure 3.2 provides a diagram for the situation mentioned above.

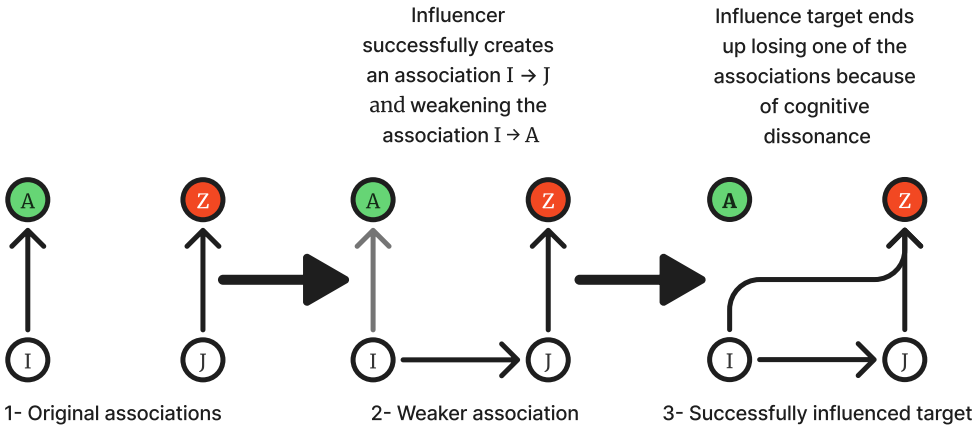

Figure 3.2: Simplified diagram showing one way an influencer may attempt to change a person's view.

The way in which these associations are created and/or removed is done through a lot of techniques, including but not limited to: appeals (appeal to fear, appeal to authority), cognitive biases (logical fallacies, cognitive dissonance, association fallacies, the tendency for conformity), demoralization, etc.

Text based propaganda detectors attempt to detect the techniques used to create or remove the associations that people have, meaning they only work during an influence operation, they are not able to work when a group of people have already been influences. Meanwhile, network based propaganda detectors take a different approach where they attempt to detect the influence agents instead of the influence technique, this is more useful from a moderation standpoint since being able to recognize the agents performing an influence operation will make it possible to reduce the impacts that these agents have.

Our technique takes an entirely different approach, where we are attempting to measure the history of associations that a community had, how these associations changed, and with the help of these changes

in associations, we might be able to deduce which influencer benefits from it, allowing us to narrow down the list of potential influencers running an influence operation.

3.1.3 Our approach

Our technique takes a paradigmatic approach, which attempts to extract meaning from the surface structure of system of signs. In machine learning terms, we will transform a collection of signs into a latent space whose axes encode different, or, a collection of different meanings.

In order to swiftly validate our approach in just a few months we have opted to only work on text data, however, it is theoretically possible to extend this approach into a multimodal system capable of accepting different sign systems, such as different languages, video, or audio, and use the same process we have used for text data, since our technique uses only the latent space, and does not rely on the sign system in use [36].

The most effective technique for converting text into a latent space that encodes meaning is using word embeddings, which are not only capable of converting similar words into a nearby cluster, but have also been shown to have the ability to encode associations between words [37], which is of significant importance to us.

Now that we have a relatively unbiased embedding, we can fine tune it on a certain community and be able to capture the associations that these communities have between words (shown in the "Results" chapter). However, we will not capture all the association, we will only capture those which are widely discussed, since signs which have not been used cannot be included in the fine-tuning dataset, and those which have not been mentioned a lot may not be able to properly encode the correct associations.

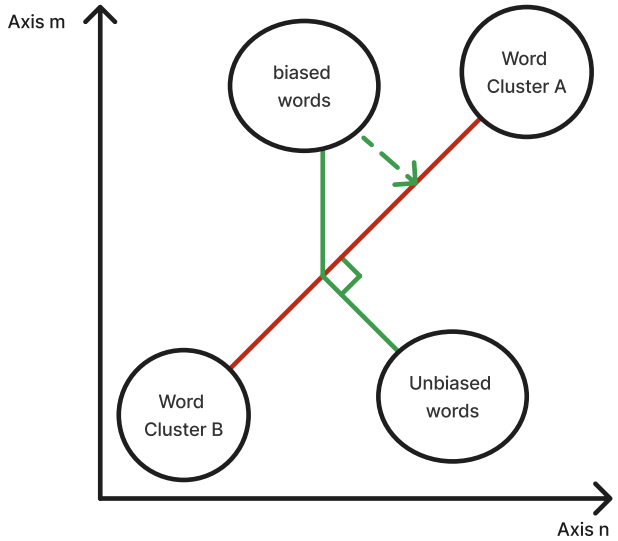

Figure 3.3: Simplified representation showing how WEAT works.

This fine-tuning step will transform the location of the words in the latent space of our embedding, all the words that exist in the dataset will shift randomly, however those that have implicit associations will tend to shift in a specific direction, the direction and magnitude of this direction can encode a large amount of information which we can use later on to extract the newly developed biases.

The associations between different words are extracted using a technique which is traditionally used for detecting biases in word embeddings, called WEAT (Word Embedding Association Tests) [38], WEAT works by defining a collection of cluster pairs, each pair encodes a certain bias, this pair allows us to get a compound axis in the latent space that is used for encoding that bias, we then measure the cosine similarity between this axis and the position of the word in the latent space, giving us a value between -1 and 1, where 0 means not biased, and 1/-1 means heavily biased in favor or in opposition of that bias (Figure 3.3 provides a simplified representation of the functioning of WEAT).

In more formal terms, WEAT defines four word clusters (X, Y, A, B), A and B are our attribute words, where A is a word cluster encoding a certain bias, and B is another word cluster encoding the opposite words of cluster A in relation to the bias, X and Y are two clusters of words which we assume are associated with one of the clusters A and B. We use these word clusters to calculate the test statistic $s(X, Y, A, B)$ where:

$$s(x, A, B) = mean_{a \in A} cos(\overrightarrow{x}, \overrightarrow{a}) - mean_{b \in B} cos(\overrightarrow{x}, \overrightarrow{b}), \tag{3.1}$$

$$s(X, Y, A, B) = \sum_{x \in X} s(x, A, B) - \sum_{y \in Y} s(y, A, B). \tag{3.2}$$

The test statistic $s(X, Y, A, B)$ in equation 3.2 will give us the magnitude of bias encoded between the word clusters X and Y, but since we are only interested in calculating the bias of a single word. We will only use the equation 3.1.

After extracting these associations we build a heatmap where one axis is used for the words, and the second axis is used for a certain concept that we choose, each cell has a value between one and zero that shows how strongly a word is associated with a concept.

Using the association heatmap we are able to perform various tasks, for example a simple approach to detecting shifts in beliefs is by getting a collection of heatmaps, where each heatmap has the associations for a specific time frame, then we monitor the associations for any rapid shifts or slow large shifts which can be a sign of a community having been influenced.

We are also able to compare the heatmap of two communities by creating a new heatmap containing the words shared by these two communities and observing the differences between the associations that they have, this may potentially allow us to recognize communities which are at a high risk of being in conflict with each other.

Table 3.1: List of words used to capture each bias.

Bias Name	Word cluster A	Word cluster B
Gender	girl, woman, female, wife, mother	boy, man, male, husband, father
Positive	good, nice, pretty, delicious, euphoria	bad, rude, ugly, disgusting, misery
Qualification	useless, unqualified, amateur, incompetent	competent, expert, qualified, skilled, useful
Humanity	civilised, human, educated, cultured	ignorant, stupid, object, animal, barbaric, inhuman, cruel, savage
Malice	malice, hate, hostile, evil, grudge, revenge	friend, respectful, friendly, respect, moral, sympathy

3.2 Training and data preprocessing

3.2.1 Dataset and preprocessing

We used the "Reddit comments corpus" dataset [39], which is a dataset containing around 1.7 billion public comments gathered using the Reddit API spanning the years 2007 to 2015, this dataset is ideal to our study for 3 main reason, first being that a large amount of communities are clearly separated, making it significantly easier for us to test our technique without the need for a step separating communities from each other resulting in extra inaccuracies, second being that Reddit allows for negative comment scores, which is useful for identifying which points the community in question strongly disagrees with, and finally, the data we are using comes from a time when bot accounts did not yet have the ability to create text that is nearly indistinguishable from human written text, allowing us to perform our study on clean, purely human comments, without the need to worry about potential biases introduced by bot accounts.

The dataset is structured in a JSON format, containing the content of a comment, and various other metadata, which include, but are not limited to: user ID, username, posting data, harvesting date, comment score, number of upvotes, number of downvotes, link to the comment, ID of the original post, etc. A large amount of this data is not of interest to us since our study is concerned with extracting biases from the text content of the comments, however we are interested in the comment score, since it allows us to filter out comments which do not align with the views of the community.

Before using the data, it first needs to go through multiple steps of cleaning and preprocessing, as a result we have created a data preprocessing pipeline, the first part involves cleaning out potential problematic comments. These include comments that have been deleted, which are easily identified by their comment content, which is: "[deleted]", after removing deleted comments, we need to ensure that the remaining comments still align with the views of our community, we do this by removing all the comments whose score is below a certain score, in most cases we set the threshold score to be 0 to ensure all the comments have a positive score, however for some trials we set the threshold to a score of 10.

It is also possible to sort the comment by descending order of their score in order to find comments which align strongly with the views of the community in relation to the rest of the comments, no special processing was done to these comments other than sorting, however it should be possible to weight the contribution of these comments by the number of upvotes it received since each upvote reflects another account that agrees with the points of that comment.

After cleaning our dataset, we need to preprocess it in order to make it usable with our model, since we are using a simple word embedding model, we need to convert the comments into the appropriate format that our word embedding uses, we opted to train our embeddings using the Word2Vec technique [40], this is a relatively old technique, however it allows us to fine tune word embeddings without the need for a specific task, it has also been shown to be more efficient than other previous techniques.

Word2Vec uses 2 types of data architecture, the first is called CBOW (Continuous Bag Of Words), and the second called Skip-gram. CBOW takes in the context in which the word appeared in, and attempts to predict the word, meanwhile Skip-gram performs the opposite, where we input the word that we want to know the embedding of, and try to predict the context in which the word appeared in. We opted to use the Skip-gram architecture in our technique, where the training objective is formally defined in equation 3.3.

$$\prod_{i \in C} Pr(w_j : j \in N + i | w_i). \tag{3.3}$$

N is the set context words, and C is the set of our corpus. We opted to use Skip-grams with a window size of 5 to fine tune our model.

Using gensim [41], we iterate through our dataset, splitting every comment into a separate array of words, this split is done after multiple comment cleaning steps in order to ensure that words are properly extracted, these steps are: 1- Add spaces between punctuation and words, 2- Split numeric characters from words, 3- convert all words into lowercase. After splitting comments into arrays, we count the number of words that occurred in our corpus, and we remove all words that do not meet a certain threshold (We used a threshold of 1 in order to remove any uncommon typos or uninteresting text).

3.2.2 Model training

We chose the GloVe word embedding [42], trained on the gigaword corpus and 2014 Wikipedia, with 200 dimensions.

Using keras we create an MLP with one hidden layer representing the embedding space, the input and output of this model equals the length of the embedding dictionary (which for this model equals 400 000), the hidden layer does not use biases and has a linear activation function, the output layer does use biases and a softmax activation function, table 3.2 shows the model used.

Using the categorical cross entropy function and the Adam optimizer, we use the skip grams from the previous step for the training process (in some occasions we reshuffle the skip grams and use the top 300k skip grams in order to expedite the training process), usually a single epoch is enough to reach a stable loss value, so for all of our tests we have not used multiple epochs.

Table 3.2: Model used for fine-tuning.

Layer name	Count	Other information
Input layer	1	/
Category encoding	400k	one hot encoding
Dense layer	200	linear activation and no biases
Output layer	400k	softmax activation

Chapter 4

Results

4.1 Quantitative results

4.1.1 Model robustness

The process of training models is a stochastic process, meaning that even if we observe a large shift in the weights of our model, that does not necessarily translate into a better model, it is also possible for our model to reach an optimal set of weights, however, as we continue training, or by slightly changing the training data, the weights could change into a less optimal position, this instability is especially common for complex tasks and architectures. Therefore, it is important for us to test the stability of our model.

A popular technique used for testing the stability and robustness of models is K-fold cross validation, this technique works by splitting our data into K chunks, and training K models such that the training set of the model n includes all chunks except for the chunk n, this is useful for verifying how stable a model is as the data used for training changes, it also lets us see how well a model will generalize to new unseen data.

We used K-fold cross validation for testing the robustness of our technique, using $K = 5$, we fetch all the posts from the month of February 2010, keep only the top 3000 comments while removing any who's score is less than 10, and creating 5 separate datasets. We fine tune the 5 embeddings and save their biases.

Using the 5 bias heatmaps, we measure the standard deviation of every bias of every word, giving us a new heatmap.

Table 4.1 shows the average standard deviation of every bias, we can see that the worst case is 0.018 which is 0.9% of all the possible values this bias can achieve, this is a decent range of possibilities, meaning we are able to trust that the values extracted by our techniques are relatively stable even with slight changes to the data used. However, we did notice that some words had a much bigger standard deviation that other words, the words that had a relatively big standard deviation were words that were uncommon in our dataset, and words which can be used in most contexts (words like: want, going, need, like, etc.), table 4.2 provides the standard deviation of the top 10 words that appeared in our training data, as we can see, words related to topics discussed in the Subreddit (government, gun, right) have a much smaller standard deviation than words which are widely used (want, think, people), the words with a large standard deviation should be easy to filter by finding the most common words of multiple communities and removing the words shared by the majority.

Table 4.1: Standard deviation of biases.

Bias	Standard deviation
Gender	0.01804
Positivity	0.01219
Incompetency	0.01408
Humanity	0.01066
Malice	0.01194

4.1.2 P value

The P value is a measure used for hypothesis testing, it helps us measure the statistical significance of our results and the likelihood that these results are caused by random noise. The P value is the

Table 4.2: Average variance of the 10 most used words.

Word	Standard deviation
people	0.01879
like	0.01251
government	0.01837
think	0.02490
want	0.02335
gun	0.01818
need	0.02284
going	0.02105
know	0.01684
right	0.01552

probability that the results found in an experiment are caused by random noise, as such, the smaller the P value, the more likely that the results that we found are caused by something else other than random noise, it is very useful in most fields of science and is usually one of the main metrics used for scientific experiments.

However, it is easy to misinterpret the P value as being a proof that the hypothesis presented is correct, because it only proves that the results are likely not caused by random noise, as a result, it is important to be careful not to misuse it as a proof of the validity of the proposed hypothesis.

We will use the P value to ensure that the biases that we extracted are statistically significant, and to also test that different communities do have noticeable different biases.

In order to do that, we follow the following steps:

1. We assume the null hypothesis "two different communities will not have different biases" and the alternative hypothesis being "communities have different biases".
2. Using the "Conservative" and "Libertarian" Subreddits, we fetch two datasets containing the comments of each community
3. We run the comments through our preprocessing pipeline and pick the top 3000 comments from both communities then shuffle the comments from both communities together.
4. Likewise, we take the first 3000 comments from the shuffled comments and train our model then save the extracted biases.
5. We repeat this process 20 times in order to minimize the effect of random noise
6. We do the same process with 20 other models that only contain comments from one of the Subreddits
7. We use the student's T test to calculate the P value for each word

We chose the two Subreddits "Libertarian" and "Conservative" because they are both political Subreddits, meaning that the words used in their comments will be more similar than if we used two unrelated Subreddits, they are also relatively different, in the sense that they both represent opposing political views, allowing us to ensure that our technique is capable of differentiating between these two.

After running the student's T test, we get a heatmap that encodes the P value of each word in each bias, this is not very understandable, so we average all the P values to get single number, since each word can be between the range of 0 and 1, we also provide in table 4.3 more information regarding the distribution of the P values throughout the entire heatmap.

Our average came out to 0.033, signifying that the majority of the words had a P value less than 0.05, which is the most common threshold used. However, each word has its own P value, which can also signify how robust the association between a word and the bias is, where a larger P value can signify a less robust association, and a smaller P value signifies a more robust association, table 4.4 shows the P values of the top 10 words used in both communities.

Table 4.3: statistically relevant values from the P value heatmap.

Average	0.03379
Minimum	6.30879e-32
Maximum	0.99942
Standard deviation	0.13542

Table 4.4: 10 most used words and their P value.

Word	P value
people	0.00035
like	0.01345
don't	0.057527
it's	0.12534
think	4.82685e-06
government	0.11197
i'm	0.018579
obama	0.0317382
going	2.12132e-05
that's	0.00100

4.1.3 Stability across different embedding models

Throughout our tests, we used one single embedding model, which is GloVe, trained on the gigaword corpus and Wikipedia, with 200 axes in the latent space. Theoretically, our technique should be able to generalize to different models with different numbers of latent spaces, but it is important to test it nonetheless to ensure that our technique is still capable of working when changing the embedding models, or in future work with multimodal and multilingual embeddings.

To do this, we used the following models: GloVe pretrained on 2B tweets with 25 dimensions, GloVe pretrained on 2B tweets with 50 dimensions, GloVe pretrained on the Gigatext corpus and Wikipedia with 50 dimensions, GloVe pretrained on the Gigatext corpus and Wikipedia with 100 dimensions, and our original model GloVe pretrained on the Gigatext corpus and Wikipedia with 200 dimensions. We were not able to use models with dimensions higher than 300 as a result of technical limitation.

Each model was fine-tuned on the same dataset using K-fold cross validation, which gave us a list of biases, we then compare these biases by calculating their standard deviation, and comparing it with the biases extracted using the different models.

Figure 4.1 shows a comparison between the extracted biases of the word "people" compared to different models, we chose this word because it has the most common occurrence in our corpus. We can see a clear separation between the models which were pretrained on the twitter corpus and those pretrained on the Gigaword corpus, a possible explanation for this is that the twitter corpus used for pretraining may be more biased since it is trained on content from social media, meanwhile the Gigaword corpus is composed mainly of content from news articles and press services, which tend to be more objective than social media posts, in other words, we could be observing biases inherent from the dataset used for pretraining the embedding model.

Figure 4.2 shows the averaged standard deviation across all words between different models, the chart contains three separate bars, the first contains the standard deviation of the models pretrained on the gigaword corpus, the second contains the models trained on the twitter dataset, and the third contains the models from both datasets. We can see that the standard deviation for models that shared the same dataset is relatively low when compared to the standard deviation from models that shared both of them, this hints that the reason these two models have significantly different biases is caused by the pretraining dataset, and is not caused by the model being unable to extract these biases.

4.2 Qualitative results

4.2.1 Effect of biases on downstream tasks

In order to see how the associations made by communities can impact their perception on the world, and how these implicit biases can be misleading, we decided to study the impacts that a fine-tuned embedding has on a sentiment analysis task compared to the base non-tuned and less biased embedding.

To do this we used of the NRC emotion lexicon dataset [43] for training, we created 2 models: one which uses the base relatively unbiased embeddings called the "base" model, and the second "biased" model that uses embeddings fine-tuned on the "Libertarian" Subreddit. Both models use the same architecture which is mentioned in table 4.5 and were compiled using the Adam optimizer and the categorical cross entropy loss function. Our dataset consisted of the "positive" and "negative" emotions only, and the words were filtered in order to only keep the words that exist in the word embedding used.

After training both models for 10 epochs, with a batch size of 32, we noticed that the accuracy of the base model stabilized at around 82%, meanwhile the accuracy of the biased model stabilized at around 65%. It is important to note that the dataset is relatively unbalanced, where there are 2313 negative labels, 1619 positive labels, 61 labeled as both positive and negative, and 6030 which have no label. However, even with this unbalance, a difference of 17% is too big to be attributed to random noise or to an unbalanced dataset, especially considering that both models were trained on the same words, and

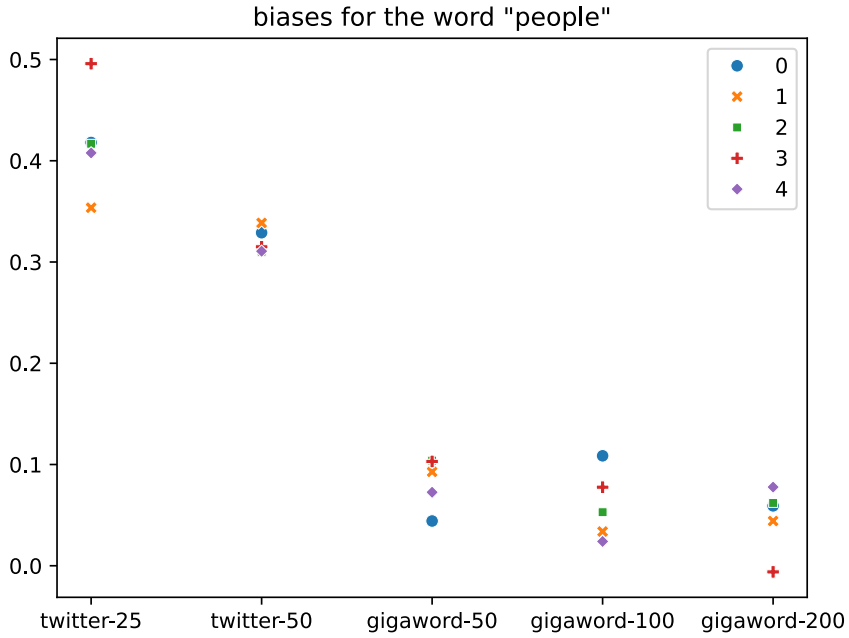

Figure 4.1: Extracted biases on the word ”people” using different embedding models.

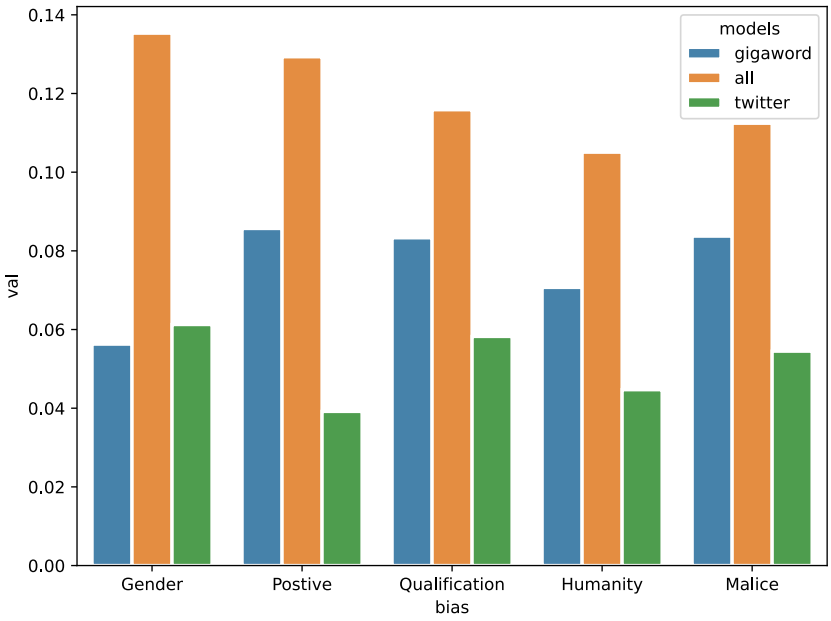

Figure 4.2: Average standard deviations across words for models trained on gigaword corpus, twitter dataset, and both.

were tested against the same words. Table 4.6 Provides a sample of the words falsely predicted by the biased model.

We believe that this drop in accuracy is caused by associations made by a biased community, where words which originally do not have any associations with being positive or negative (for example in a political context the words ”left” and ”right”) develop these associations as a community mostly uses them in a positive or a negative context, this use in biased contexts shifts the location of this word in the latent space such that the new location of that word is now located in a more positive or negative location relative to the bias projection axis.

Table 4.5: Sentiment analysis model architecture.

Layer name	count	other info
Input layer	200	/
Dropout layer	/	0.2 rate
Dense layer	128	ReLu activation
Dropout layer	/	0.2 rate
Output layer	2	softmax

Table 4.6: Sample of random 10 words and their predictions.

Word	Predicted label	True label
turn	72% Positive	Neutral
note	99% Negative	Neutral
debt	98% Positive	Neutral
federal	80% Positive	Neutral
eight	76% Positive	Neutral
version	92% Negative	Neutral
certainly	99% Positive	Neutral
zealand	97% Positive	Neutral
bar	99% Positive	Neutral
victory	87% Negative	Neutral

4.2.2 Inter-community differences

Our work focuses on identifying the problematic associations made by different communities, so, we need a way to easily visualize these associations in a manner which is easily understandable, luckily, we only have a small number of biases (6 in total), and the association between each word and bias is a single number between -1 and 1, allowing us to plot them in a bar chart. This does introduce a different problem, which is the selection criteria of which words do we show (our embedding model has a dictionary of 400k words), unfortunately there is no easy solution for this and the words must either be filtered to fit certain criteria, or manually selected, we chose the first option, where we only kept the most used words to avoid those with unstable or inaccurate biases, after that, each bias was separated and the words were sorted in descending order according to how big the difference was between them and the other community.

The two communities used were the ”Conservative” and ”Libertarian” Subreddits from the month February 2010, and the fine-tuning was done in the same manner that was used for calculating the P value, the top 3000 comments with the highest score were kept, and their skip grams were used to fine-tune the embedding.

The figures 4.3, 4.4, 4.5, 4.6, and 4.7 show the list of the words selected for each bias, and how strong the association is for both communities, red was chosen for the ”Conservative” Subreddits and blue for the ”Libertarian” Subreddit.

We can see some interesting associations in these plots, for example when it comes to the words ”cops” and ”state”, we can see that they are seen to be more friendly in the ”Conservative” Subreddit when compared to the ”Libertarian” Subreddit.

There are many other associations to be seen from these plots, however, we do see that there is some noise as a result of our word selection criteria, more specifically, pronouns and words which are not related to the topics discussed in these communities, these are harder to interpret the meaning of since they could point at a multitude of other words, or a single word depending on the context in which they appeared in, this could potentially be resolved by replacing these words with the words that they are referencing, as for general words, more effort could go into understanding what these associations mean, or we can simply add them to the stop list in order to minimize the noise caused by them.

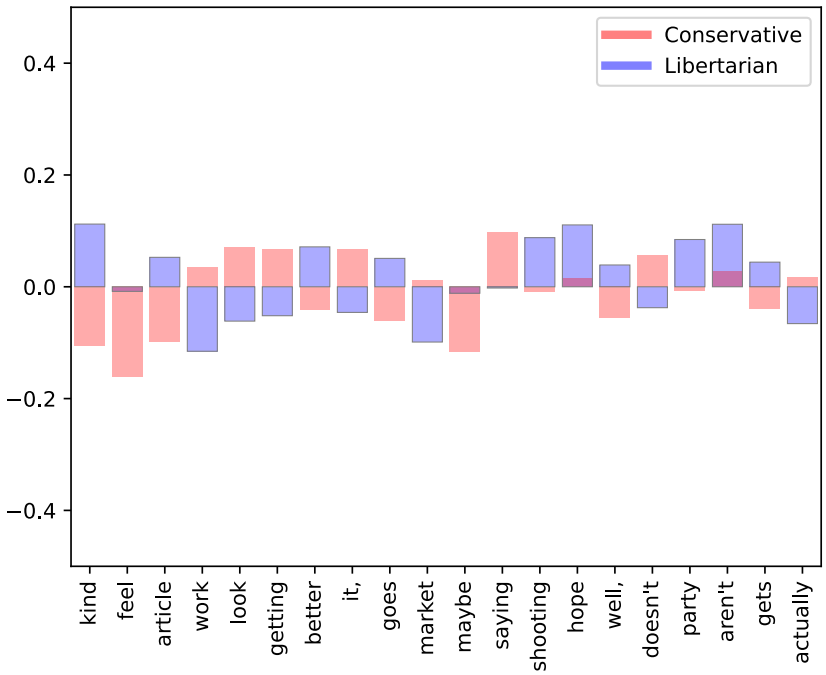

Figure 4.3: Gender bias in words where positive is female and negative is male.

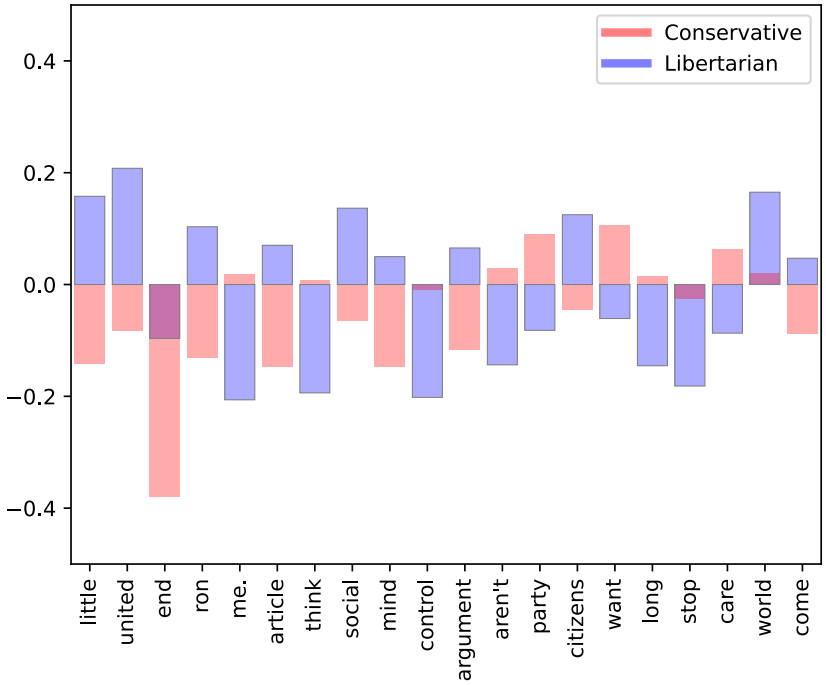

Figure 4.4: Humanity bias in words where positive is human and negative is inhuman.

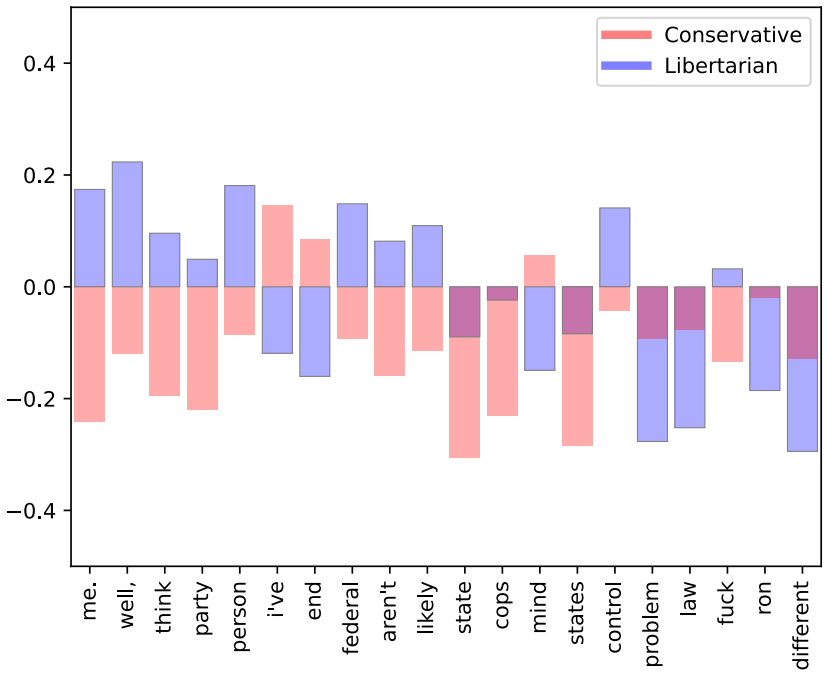

Figure 4.5: Malice bias in words where positive is malicious and negative is friendly.

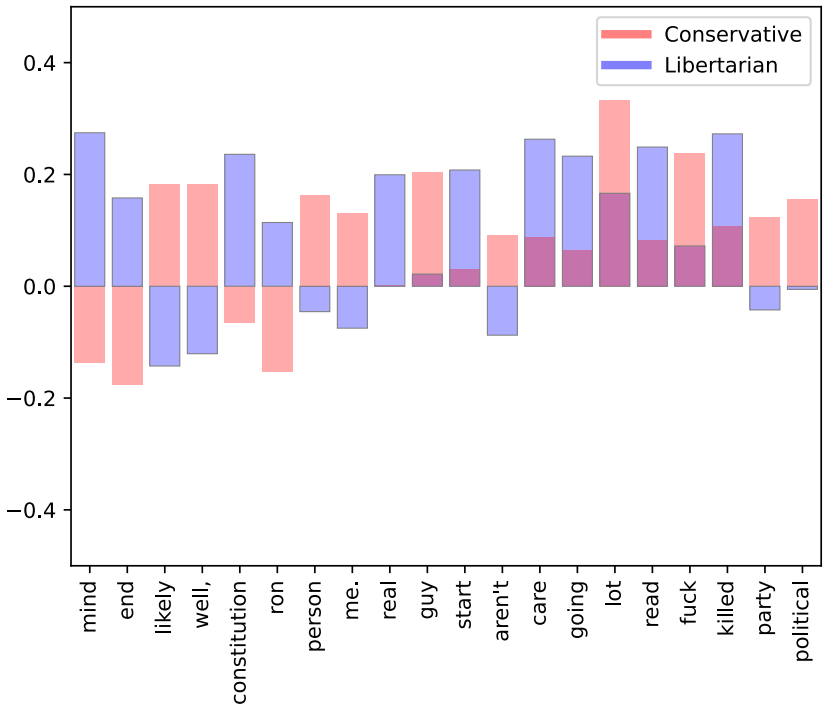

Figure 4.6: Positive bias in words where positive is positive and negative is negative.

4.2.3 Changes over time

Finally, our work focuses on identifying shifts in associations, as a result, it is important to test how well our technique is at detecting those shifts over time.

To do so we extracted the biases from the "Libertarian" Subreddit throughout the year 2010, we did so in the same manner to how we calculated the P value, by ordering the comments by their scores in descending order, using the top 3000 comments, preprocessing them, fine-tuning our model, and measuring the biases.

Each month was measured separately, then the results were aggregated into a single matrix, which we

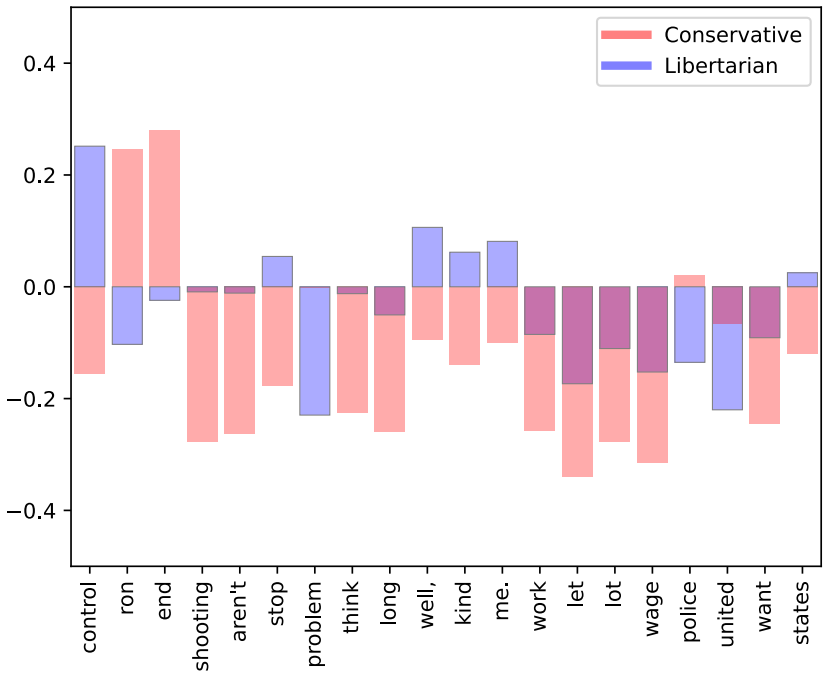

Figure 4.7: Qualification bias in words where positive is unqualified and negative is qualified.

later used with the help of a simple 3σ anomaly detection approach, since the amount of points is only 12 (one point for each month), it is expected for the anomaly detection to not be very successful, which is why we also manually selected a few words we assumed to be of interest.

Figures 4.8, 4.9, and 4.10 show the bias history of the three words ”congress”, ”obama”, and ”senate”. We can see clear sudden shifts in biases during some months, even though our technique does not provide any technique to verify the source of this shift, we can check certain significant events that happened during, or before that month, those events are outlined in the figures as potential causes for these sudden shifts.

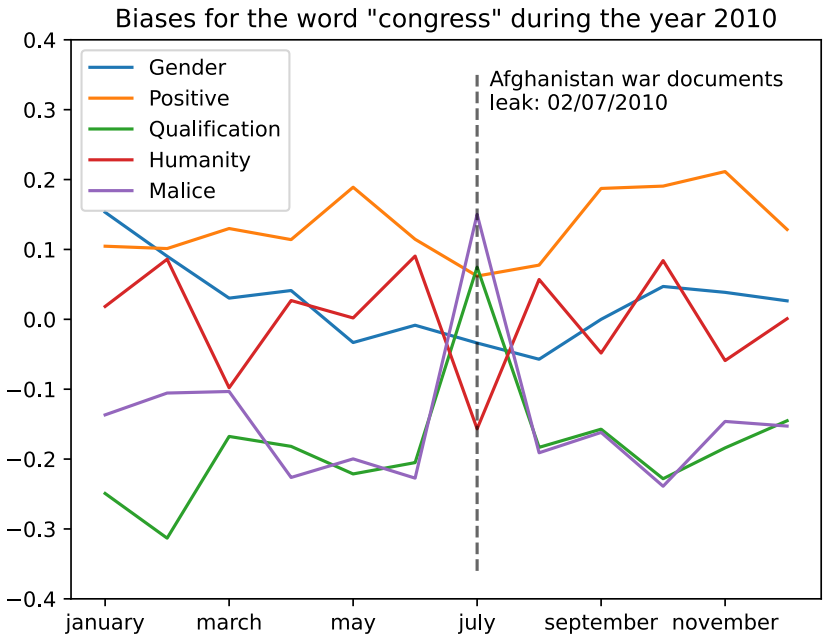

Figure 4.8: The history of biases for the word ”congress” spanning all the months of 2010.

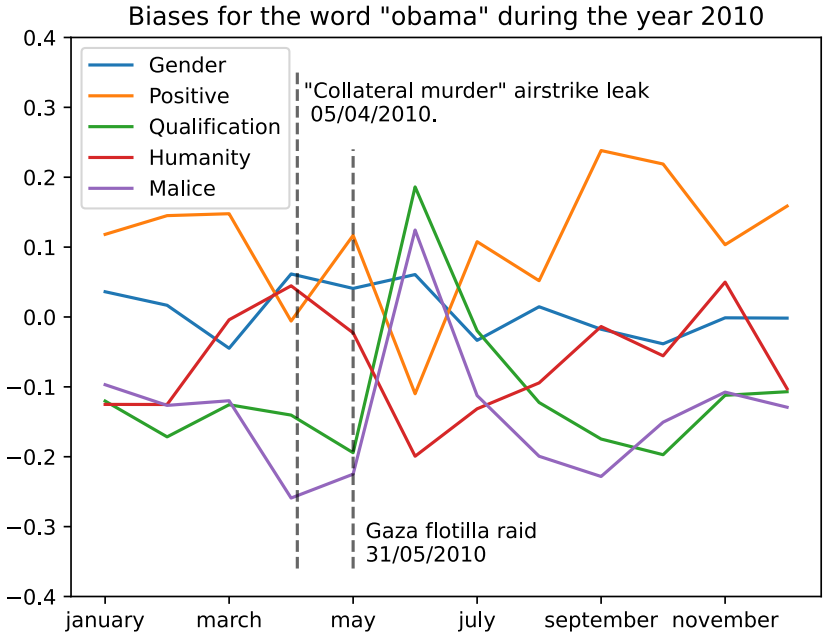

Figure 4.9: The history of biases for the word ”senate” spanning all the months of 2010.

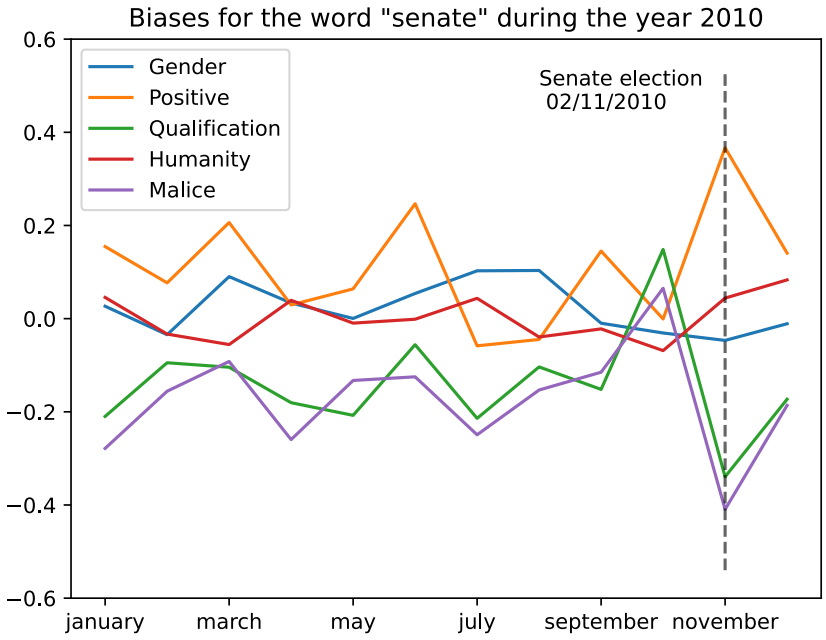

Figure 4.10: The history of biases for the word ”senate” spanning all the months of 2010.

Chapter 5

Conclusion

5.1 Limitations

Even though our technique does overcome many of the limitations faced by most previous works, it still suffers from many limitations which should be considered when using it.

The most obvious limitation has to be the inability for our technique to extract associations for topics which the community does not mention or talk about often, it may be possible to deduce these associations by comparing the community of interest with other communities that have similar associations, however it is still a significant downside since there is no way of passively verifying the validity of these associations with our community.

Our technique also assumes that all comments in the dataset have the same level of popularity, and the score of each comment is not taken into account when calculating the associations, this is problematic since there could be topics which a community does not have a solid agreement on, and as such tracking the score of each post could provide more useful insight into which topics the community of interest strongly agree on, and which topics are controversial.

And as noted before, in order to reliably extract the associations created by a community, we need to first aggregate the comments made by that singular community, this is easy when it comes to performing studies on social media platforms, since communities are explicitly stated, even when they are not, it is relatively easy to group users into communities by studying the relationship graphs of these accounts, the difficulty is more prominent when using mediums that do not have clear separation between communities, such as blogs, news sites, text boards, etc.

It is also not designed to detect an influence operation as it is happening in real time, even though it is theoretically possible to do so when the influence actors are known. We believe this is an appropriate trade-off in return to being able to get a detailed description of the effects of an influence operation.

Finally, this technique uses a significant amount of computational power, for example, fine-tuning the 20 models for calculating the P value (where the amount of skip grams was capped to 300k as opposed to the original 3 million) took around 10 hours of computing on an RTX-3060. If we are interested in performing a large scale study on thousands of communities (that each have thousands or millions of posts), we will end up with an unfeasible amount of time, as such further research should prioritize finding a more efficient, and less energy intensive approach at extracting associations, that does not involve fine-tuning large models.

5.2 Future work

There are many points that can be improved by future research, either in order to improve the efficacy of the technique, provide more explainable results, or even use it to create datasets for future research, we identified a few points that we believe could be improved.

First is using TF-IDF (Term Frequency, Inverse Document Frequency) in order to use as weight for the biases during the process of extraction, this could potentially make the process more robust, and make it more clear which biases have shifted significantly, this is because during our work, we noticed that some words which have not had many occurrences, ended up having an unstable significant shift in biases, this could potentially be attributed to the number of occurrences of the word resulting in it appearing in only some contexts, resulting in a biased view of that word.

Another improvement can be done to enhance the explainability of our technique, is by using contextual embeddings, this can allow us to differentiate between homographs and polysemy, also, it may be possible to use the attention mechanism's K and Q matrices to find more associations integrated into the embedding model, however the latter is significantly more difficult than the former.

When we defined our problem, we made sure to use definitions which do not specify the modality of our data, this was done on purpose since our techniques could still be applied using multilingual or multimodal embeddings, it should be pretty easy to apply the process we applied, creating useful bias

axes for multimodal embeddings could prove challenging, however we believe the biases used for text data could be sufficient for this task.

Finally, it is possible to use the bias history of multiple communities to create datasets, which can be used to train models to predict an influence operation as it is occurring, or to predict the reaction of a community to certain content.

5.3 Ethical considerations

It is clear that the topic regarding influence operations is a sensitive one, and generally conversations regarding it are seen as taboo, not to mention that many fields such as public relations have many heated debates on the ethics of performing influence operations.

We believe that our work does not violate any ethical guidelines, since we have attempted to be as transparent as possible on the type of data, models, and techniques used. We also aim to identify influence operations, rather than perform them, which we believe is ethical.

Bibliography

[1] D. Brendan Nagle and Stanley Mayer Burstein. *The Ancient World Reader: Readings in Social and cultural history*. Prentice Hall; Pearson Education distributor, 2009.

[2] Bruce Lannes Smith. propaganda. Encyclopedia Britannica, February 2025.

[3] Christian Staal Bruun Overgaard. Perceiving affective polarization in the united states: How social media shape meta-perceptions and affective polarization. *Social Media + Society*, 10(1):20563051241232662, Febraury 2024.

[4] Giovanni Spitale, Nikola Biller-Andorno, and Federico Germani. Ai model gpt-3 (dis)informs us better than humans. *Science Advances*, 9(26):eadh1850, June 2023.

[5] Giovanni Da San Martino, Stefano Cresci, Alberto Barron-Cedeno, Seunghak Yu, Roberto Di Pietro, and Preslav Nakov. A survey on computational propaganda detection, July 2020.

[6] Kilian Sprenkamp, Daniel Gordon Jones, and Liudmila Zavolokina. Large language models for propaganda detection, November 2023.

[7] Akib Mohi Ud Din Khanday, Mudasir Ahmad Wani, Syed Tanzeel Rabani, Qamar Rayees Khan, and Ahmed A Abd El-Latif. HAPI: An efficient hybrid feature engineering-based approach for propaganda identification in social media. *PLoS One*, 19(7):e0302583, July July 2024.

[8] V.S. Subrahmanian, Amos Azaria, Skylar Durst, Vadim Kagan, Aram Galstyan, Kristina Lerman, Linhong Zhu, Emilio Ferrara, Alessandro Flammini, and Filippo Menczer. The darpa twitter bot challenge. *Computer*, 49(6):38–46, June June 2016.

[9] G. Da San Martino, A. Barrón-Cedeño, H. Wachsmuth, R. Petrov, and P. Nakov. Semeval-2020 task 11: Detection of propaganda techniques in news articles, September 2020.

[10] Media bias/fact check (mbfc). https://mediabiasfactcheck.com/.

[11] Malak Abdullah, Ola Altiti, and Rasha Obiedat. Detecting propaganda techniques in english news articles using pre-trained transformers. In *2022 13th International Conference on Information and Communication Systems (ICICS)*, pages 301–308, June 2022.

[12] Alexander Chernyavskiy, Dmitry Ilvovsky, and Preslav Nakov. Unleashing the power of discourse-enhanced transformers for propaganda detection. In Yvette Graham and Matthew Purver, editors, *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1452–1462, St. Julian's, Malta, March March 2024. Association for Computational Linguistics.

[13] Jakub Piskorski, Nicolas Stefanovitch, Giovanni Da San Martino, and Preslav Nakov. SemEval-2023 task 3: Detecting the category, the framing, and the persuasion techniques in online news in a multi-lingual setup. In Atul Kr. Ojha, A. Seza Doğruöz, Giovanni Da San Martino, Harish Tayyar Madabushi, Ritesh Kumar, and Elisa Sartori, editors, *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*, pages 2343–2361, Toronto, Canada, July July 2023. Association for Computational Linguistics.

[14] Europe media monitor - newsbrief. http://emm.newsbrief.eu/.

[15] Timo Hromadka, Timotej Smolen, Tomas Remis, Branislav Pecher, and Ivan Srba. Kinitveraai at semeval-2023 task 3: Simple yet powerful multilingual fine-tuning for persuasion techniques detection. *arXiv preprint arXiv:2304.11924*, April 2023.

[16] Ben Wu, Olesya Razuvayevskaya, Freddy Heppell, João A Leite, Carolina Scarton, Kalina Bontcheva, and Xingyi Song. Sheffieldveraai at semeval-2023 task 3: Mono and multilingual approaches for news genre, topic and persuasion technique classification. *arXiv preprint arXiv:2303.09421*, March 2023.

[17] Antonio Purificato, Roberto Navigli, et al. Apatt at semeval-2023 task 3: the sapienza nlp system for ensemble-based multilingual propaganda detection. In *Proceedings of the The 17th International Workshop on Semantic Evaluation (SemEval-2023)*. Association for Computational Linguistics (ACL), July 2023.

[18] WILLIAM C. MANN and SANDRA A. THOMPSON. Rhetorical structure theory: Toward a functional theory of text organization. *Text - Interdisciplinary Journal for the Study of Discourse*, 8(3):243–281, 1988.

[19] Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. Deberta: Decoding-enhanced bert with disentangled attention, October 2021.

[20] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. Unsupervised cross-lingual representation learning at scale, April 2020.

[21] Alberto Barrón-Cedeño, Israa Jaradat, Giovanni Da San Martino, and Preslav Nakov. Proppy: Organizing the news based on their propagandistic content. *Information Processing and Management*, 56(5):1849–1864, May 2019.

[22] Alberto Barrón-Cedeno, Giovanni Da San Martino, Zhang Yifan, Ahmed Ali, Fahim Dalvi, et al. Qlusty: Quick and dirty generation of event videos from written media coverage. In *CEUR Workshop Proceedings*, volume 2079, pages 27–32. CEUR-WS, March 2018.

[23] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In *kdd*, volume 96, pages 226–231, 1996.

[24] Paul Jaccard. Étude comparative de la distribution florale dans une portion des alpes et des jura. *Bull Soc Vaudoise Sci Nat*, 37:547–579, 1901.

[25] Clayton Allen Davis, Onur Varol, Emilio Ferrara, Alessandro Flammini, and Filippo Menczer. Botornot: A system to evaluate social bots. In *Proceedings of the 25th International Conference Companion on World Wide Web*, WWW '16 Companion, page 273–274, Republic and Canton of Geneva, CHE, April 2016. International World Wide Web Conferences Steering Committee.

[26] Kyumin Lee, Brian Eoff, and James Caverlee. Seven months with the devils: A long-term study of content polluters on twitter. In *Proceedings of the international AAAI conference on web and social media*, volume 5, pages 185–192, August 2011.

[27] Shangbin Feng, Herun Wan, Ningnan Wang, Zhaoxuan Tan, Minnan Luo, and Yulia Tsvetkov. What does the bot say? opportunities and risks of large language models in social media bot detection, February 2024.

[28] Shangbin Feng, Herun Wan, Ningnan Wang, Jundong Li, and Minnan Luo. Twibot-20: A comprehensive twitter bot detection benchmark. In *Proceedings of the 30th ACM international conference on information & knowledge management*, pages 4485–4494, June 2021.

[29] Shangbin Feng, Zhaoxuan Tan, Herun Wan, Ningnan Wang, Zilong Chen, Binchi Zhang, Qinghua Zheng, Wenqian Zhang, Zhenyu Lei, Shujie Yang, et al. Twibot-22: Towards graph-based twitter bot detection. *Advances in Neural Information Processing Systems*, 35:35254–35269, June 2022.

[30] Zhaoxuan Tan, Shangbin Feng, Melanie Sclar, Herun Wan, Minnan Luo, Yejin Choi, and Yulia Tsvetkov. Botpercent: Estimating bot populations in twitter communities, February 2023.

[31] Jason Koebler. Researchers secretly ran a massive, unauthorized ai persuasion experiment on reddit users. *404media*, April 2025.

[32] F. Heylighen. Memetics. In *Principia Cybernetica Web (Principia Cybernetica, Brussels)*, page url: http://pespmc1.vub.ac.be/MEMES.html, created: 1993, Modified: 2001.

[33] S. E Asch. Effects of group pressure upon the modification and distortion of judgments. *H. Guetzkow (Ed.)*, Groups, leadership and men; research in human relations (pp. 177–190). Carnegie Press., 1951.

[34] Christopher J. Burke, Philippe N. Tobler, Wolfram Schultz, and Michelle Baddeley. Striatal bold response reflects the impact of herd information on financial decisions. *Frontiers in Human Neuroscience*, Volume 4 - 2010, June 2010.

[35] Dedy Darsono Gunawan and Kun-Huang Huarng. Viral effects of social network and media on consumers' purchase intention. *Journal of Business Research*, 68(11):2237–2241, June 2015.

[36] Christopher Olah. Deep learning, nlp, and representations. https://colah.github.io/posts/2014-07-NLP-RNNs-Representations/, July 2014.

[37] Kawin Ethayarajh, David Duvenaud, and Graeme Hirst. Understanding undesirable word embedding associations, August 2019.

[38] Aylin Caliskan, Joanna J. Bryson, and Arvind Narayanan. Semantics derived automatically from language corpora contain human-like biases. *Science*, 356(6334):183–186, April April 2017.

[39] u/Stuck_In_the_Matrix. "i have every publicly available reddit comment for research. 1.7 billion comments @ 250 gb compressed. any interest in this?". https://www.reddit.com/r/datasets/comments/3bxlg7, July 2015.

[40] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space, January 2013.

[41] Radim Řehůřek. Gensim - topic modeling for humans. https://radimrehurek.com/gensim/index.html, August 2024.

[42] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, October 2014.

[43] Saif Mohammad and Peter Turney. Emotions evoked by common words and phrases: Using mechanical turk to create an emotion lexicon. In *Proceedings of the NAACL HLT 2010 workshop on computational approaches to analysis and generation of emotion in text*, pages 26–34, June 2010.