TU WIEN Informatics

Propaganda Detection in Russian and American News Coverage about the War in Ukraine through Text Classification

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieur

im Rahmen des Studiums

Business Informatics

eingereicht von

Vitalij Hein, B.Sc.
Matrikelnummer 11932447

an der Fakultät für Informatik

der Technischen Universität Wien

Betreuung: Ao.Univ.Prof. Mag. Dr. Dieter Merkl

Wien, 6. August 2023

Vitalij Hein

Dieter Merkl

Informatics

Propaganda Detection in Russian and American News Coverage about the War in Ukraine through Text Classification

DIPLOMA THESIS

submitted in partial fulfillment of the requirements for the degree of

Diplom-Ingenieur

in

Business Informatics

by

Vitalij Hein, B.Sc.
Registration Number 11932447

to the Faculty of Informatics

at the TU Wien

Advisor: Ao.Univ.Prof. Mag. Dr. Dieter Merkl

Vienna, 6th August, 2023

Vitalij Hein

Dieter Merkl

Erklärung zur Verfassung der Arbeit

Vitalij Hein, B.Sc.

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Wien, 6. August 2023

Vitalij Hein

Acknowledgements

After spending a year immersed in my thesis work, I want to acknowledge those who made this journey possible.

Ao.Univ.Prof. Mag. Dr. Dieter Merkl, my supervisor, has been an incredible guiding force throughout this time. He patiently helped me navigate through the complexities of scientific work, always knowing when to apply a gentle push to keep me motivated while respecting my personal work rhythm. I would also be remiss not to mention his invaluable travel advice for Italy.

My heartfelt thanks goes to Vanessa Raffler as well. Her assistance, filled with valuable ideas and improvements, was crucial while developing the Propaganda Detection Model. I am particularly grateful for her patient ear when I was burdened with concerns. Her presence made the process significantly less daunting.

Lastly, I owe an immeasurable debt of gratitude to my parents. Their unwavering emphasis on the importance of education has always inspired us to strive for more in life. Their support instilled a drive within us to reach for our ambitions, and for that, I am eternally grateful.

Abstract

During different decades, propaganda was a vital technique to manipulate human opinions subtly. With the rise of mass media, subtle opinions can be incepted by various sources without being detected by the reader. A whole population can be manipulated into believing certain aspects this way. This thesis investigates the issue of propaganda in news articles. By applying the CRISP-DM research methodology, the study analyzes propaganda techniques in Russian and American news contexts, focusing on media coverage of the attack on the Ukrainian. The research presents a novel Propaganda Detection Model that utilizes Ensemble Learning. The model combines multiple Transformer Models, including BERT, OPT, and RoBERTa, to detect 18 propaganda techniques. A key finding is the effectiveness of Meta Classification with Model Stacking, surpassing Ensemble Averaging and other methods, achieving a Micro F1-Score of 0.78, indicating a significant level of accuracy in Propaganda Detection. The research reveals the potential of the Propaganda Detection Model to counteract propaganda in news articles, contributing to preserving democratic processes. It contributes to understanding the utilization of propaganda in Russian and American news contexts by analyzing its monthly usage from January 2022 to April 2023 and comparing the distribution of propaganda techniques.

Kurzfassung

In der Vergangenheit war Propaganda eine entscheidende Technik, um die Meinungen von Menschen subtil zu manipulieren. Mit dem Aufstieg der Massenmedien können subtile Meinungen von verschiedenen Quellen vermittelt werden, ohne dass der Leser dies erkennt. Auf diese Weise kann eine ganze Bevölkerung dazu manipuliert werden, bestimmte Aspekte zu glauben. Diese Thesis untersucht das Thema Propaganda in Nachrichtenartikeln. Durch Anwendung der CRISP-DM-Forschungsmethodik analysiert die Studie Propaganda-Techniken im russischen und amerikanischen Nachrichten, mit Fokus auf die Medienberichterstattung über den Angriff auf die Ukraine. Die Forschung präsentiert ein Propaganda Detection Model, das Ensemble Learning verwendet. Das Modell kombiniert mehrere Transformer Models, einschließlich BERT, OPT und RoBERTa, um 18 Propaganda-Techniken zu erkennen. Eine zentrale Erkenntnis ist die Wirksamkeit von Meta Classification mit Model Stacking, was Ensemble Averaging und andere Methoden übertrifft und einen Micro F1-Score von 0,78 erreicht, was auf ein hohes Maß an Genauigkeit bei der Propaganda-Erkennung hindeutet. Die Forschung zeigt das Potenzial des Propaganda Detection Model, Propaganda in Nachrichtenartikeln entgegenzuwirken und damit demokratische Prozesse zu bewahren. Sie trägt zum Verständnis der Nutzung von Propaganda im russischen und amerikanischen Nachrichtenkontext bei, indem sie deren monatliche Nutzung von Januar 2022 bis April 2023 analysiert und die Verteilung der Propaganda-Techniken vergleicht.

Contents

CHAPTER 1

Introduction

1.1 Motivation & Problem Statement

With the rise of digitalization, news consumption changed totally. The Digital News Report [NFR+22] states that traditional media consumption, like TV and print, declined further in 2021, but online and social consumption do not fill the gap. News consumption is declining, and the overall interest in news is also falling across the investigated markets. More precisely, the interest has fallen from 63% in 2017 to 51% in 2022 [NFR+22].

According to an EU study about disinformation and propaganda, updated in 2021, recent developments in disinformation campaigns nowadays do not care about the true/false dichotomy. The goal of modern disinformation campaigns is not to convince their target audience of alternative narratives, but instead, they aim to create or deepen social division. Many campaigns aim to pit social groups against each other by distributing polarised content tailored to both sides. Also, distrust is created by putting forward so-called alternative facts. The growing distrust towards media, news, and politics can have far-reaching consequences for democracy, like reduced civic participation in the political process or even decreased acceptance of democratic legitimacy in the eyes of citizens [BHL+21].

Modern propaganda works with the intelligent use of images and symbols to appeal to human biases and emotions. Propaganda is a communication technique that makes a recipient believe a foreign opinion, delimiting from persuasion: The recipient is manipulated to believe an opinion that would be unacceptable under other circumstances [PA01].

With the Russian attack on Ukraine, Russian propaganda became a broadly discussed topic in Germany. Its interest can be shown with the help of Google Trends[1]. Google

[1] https://trends.google.com (last accessed: 29 July 2022)

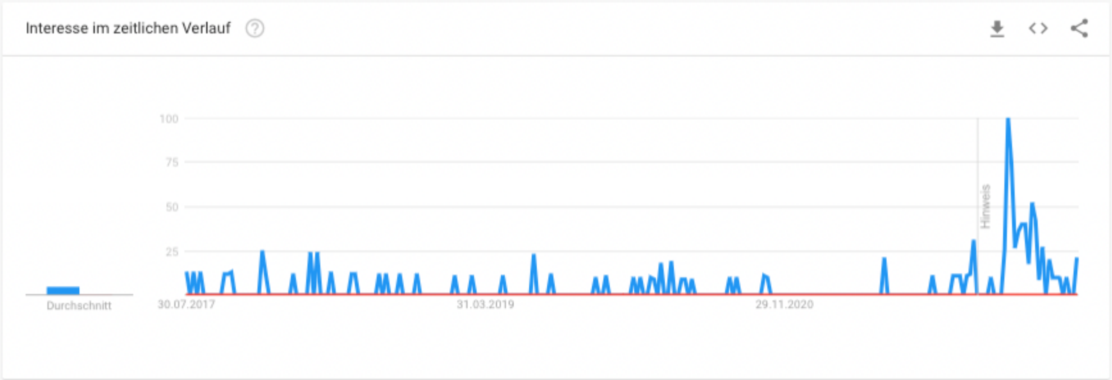


Figure 1.1: Interest in American and Russian propaganda in Germany from 2017 showing the consequent 5 years. Data source: Google Trends

Trends aggregates all search requests that their users do. All the distinct search requests are then split into coherent topics. By checking Google Trends, interest in topics can be evaluated. Interest, in this case, is the aggregated number of search requests for a specific topic. Interest is indicated in a range between 0-100 for any given date. When checking Google Trends with the search strings "Russia Propaganda" and "America Propaganda", "5 years from now" and only in "Germany" reveals an interesting fact: As shown in Fig. 1.1, the interest in American propaganda is indicated as non-existent by Google Trends, without any further data displayed. In contrast, interest in Russian propaganda had a massive spike around February 2022 and slowly declined to lower relevancy. The blue line represents an interest in Russian Propaganda, while the red line represents American Propaganda.

Based on this observation, propaganda is a phenomenon that is important in the context of Russia but not in the context of the USA. However, does this mean that states like the USA do not use propaganda techniques? Since Google Trends only analyzes search requests, it only shows the collective focus and reveals potential unconscious bias in the thinking processes of a group or society.

1.2 Research Questions

The following results are expected to be answered when researching the utilization of propaganda techniques in Russian and American news.

1.2.1 In which way can the findings of the SemEval 2020 Task 11 [DSMBCW+20] be combined, and to which degree in terms of F1-Score will this combination be able to compete with the results of the Technique Classification subtask?

Since about 25 teams published a paper, many different approaches were tested, and therefore combining different ideas and approaches could lead to an even better model. By examining those approaches, a Propaganda Detection Model is created and gradually enhanced with strategies from the other teams.

1.2.2 What are the similarities and differences in using the different propaganda techniques when looking at Russian and American news articles?

After creating a Propaganda Detection Model, the model will be used against an unlabelled, mined dataset of articles from Russian and American news sources. The model should be able to classify the sentences of the different news articles and show if a propaganda technique is used there. It is possible to show which techniques both countries use and to what extent. For example, a possible outcome could be that American articles utilize *Whataboutism* more extensively than Russian articles. In contrast, Russians could use *Reductio ad Hitlerum* more often since they claim to denazify Ukraine [Heb22]. Such understanding can lead to a better understanding of disinformation and the usage of propaganda, which is an essential suggestion in European research on disinformation and propaganda [BHL+21].

1.2.3 How did the usage of different propaganda techniques change during the timeline of the Ukrainian war?

There are 18 different propaganda techniques the model can classify. The scraped data from Russian and American news outlets contains a publishing date for every article. This publishing date helps to sort the classified propaganda techniques based on their release date. It is expected to see a different distribution of the techniques during the war proceeding.

1.3 Methodology and Approach

The CRISP-DM methodology is used to build the Propaganda Detection Model.

CRISP-DM, short for Cross Industry Standard Process for Data Mining, addresses Data Mining as a creative process that requires several different skills and knowledge. Before CRISP-DM, there was no standard framework for Data Mining projects. The methodology defines an explicit process model that provides a framework for such projects. It is described in terms of the hierarchical process model, comprising four abstraction levels, ranging from general to specific. It is built upon phases, generic tasks, specialized

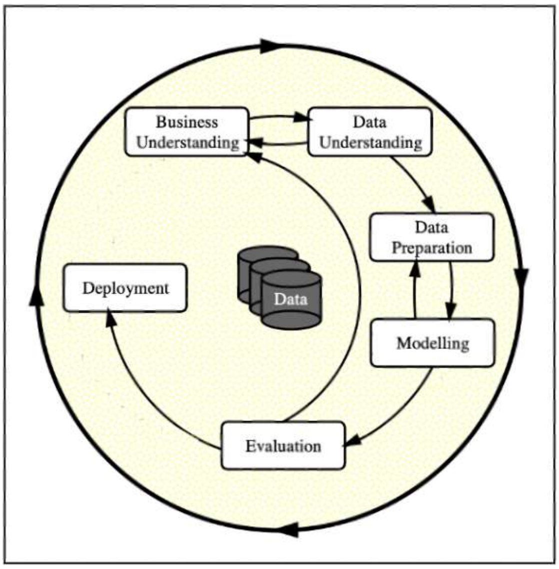


Figure 1.2: The CRISP-DM Lifecycle [WH00]

tasks, and process instances. The CRISP-DM Reference Model gives an overview of the essential phases, their tasks, and their expected output [WH00]. While the whole CRISP-DM process is extensive, some parts are skipped during the practical part since they do not apply to the thesis.

1.3.1 Business Understanding

The first phase in CRISP-DM is Business Understanding, which aims to gather a first understanding of the problem. Since a business does not instruct this thesis, the disinformation and propaganda study conducted by the European Parliament [BHL+21] is used to gather Business Understanding.

1.3.2 Data Understanding

The second phase of CRISP-DM is dedicated to Data Understanding. Data is split into the Propaganda Technique Corpus [DSMYBC+19], and the Analysis Dataset. This phase hosts statistical analysis of the datasets.

1.3.3 Data Preparation

During Data Preparation, the datasets are prepared for the consequent modeling phase, and different features are generated.

1.3.4 Modeling

The Modeling phase is all about building the Propaganda Detection Model. First, the Modeling Technique is selected, the Test Design generated, the Model Built, and the Quality Assessed.

1.3.5 Evaluation

The evaluation will occur on the Micro F1-Score since the Propaganda Technique Corpus has an uneven class distribution and a balance between Precision and Recall is needed. Also, the SemEval challenge used the Micro F1-Score as their prioritized measure for performance [DSMBCW$^+$20].

1.3.6 Deployment

The last phase in CRISP-DM is the Deployment phase, where the Deployment, Monitoring, and Maintenance are planned. The model will be made public by deploying it to Github[2]. This way, further research is encouraged.

1.4 Structure of Thesis

The first chapter briefly introduces Propaganda Detection's importance, the Research Questions, and the chosen Research Method. Chapter 2 explores the Theoretical Backgrounds of Deep Learning with a focus on Transformer-based Language Models, explaining the key concepts and theories that serve as the foundation for Literature Analysis in Chapter 3, where the SemEval 2020 Task 11 findings are summarized and analyzed. Chapter 4 then documents the execution of the CRISP-DM methodology, while in Chapter 5, the results of the three Research Questions are disclosed. The last chapter hosts a Discussion on the retrieved results of the Propaganda Detection Model, revealing Limitations and Future Research directions and an additional experiment on Russian-language articles.

[2] https://github.com (last accessed: 08 August 2022)

CHAPTER 2

Theoretical Backgrounds

This chapter serves as an overview of important theoretical concepts covered in this work.

2.1 SemEval 2020 Task 11: Propaganda Detection

The SemEval 2020 Task 11 offers a fine-grained analysis of propaganda in news articles. The challenge was split into two parts, Span Identification and Technique Classification. Further, the challenge was divided into two phases. Only training and development data were available during the first phase, while no propaganda labels were provided for the development data. The participants tried to achieve the best performance on the development data. The participants could make unlimited submissions on the development dataset, seeing the impact of their modifications and their performance compared to other participants. Gold labels were released during the second phase for the development data, and a third test dataset was released. This time the participants did not get any feedback on their performance on the test data. After the challenge was finished, only the submission system of the first phase was left open [DSMBCW$^+$20].

2.1.1 Span Identification and Technique Classification Tasks

Since the initial goal of the challenge was first to identify propagandistic spans in a sentence and second to classify them into a given set of labels, the challenge was organized into two subtasks. In the Span Identification subtask, participants were given a plain-text document and had to identify the fragments containing propaganda. In the Technique Classification subtask, text snippets identified as propaganda were already given. The participants had to build a Multi-class Classification model to predict the propaganda technique of the given text span [DSMBCW$^+$20].

2.1.2 Provided Dataset

The hosts created a labeled dataset with 18 distinct propaganda techniques annotated. The training and development datasets include 446 articles and over 400.000 tokens from 48 news outlets. The classes *Whataboutism*, *Straw Man*, and *Red Herring* were merged into one class since each is underrepresented. The three techniques share similarities since each tries to confuse a reader by bringing Attention to something irrelevant, away from the actual problem. Also, *Bandwagon* and *Reductio ad Hitlerum* were merged, both trying to approve or disapprove of an action or idea by pointing to something unpopular or popular [DSMBCW+20].

2.1.3 Propaganda Techniques

For a quick overview, the following part summarizes each propaganda technique defined by Da San Martino et al. [DSMBCW+20].

1. **Loaded Language:** Using strong emotional words to influence an audience.
2. **Name Calling or Labeling:** Labeling a subject of interest as something the audience hates or loves to connect the emotion to the subject of interest.
3. **Repetition:** Repeating a message multiple times until the message sticks with the target audience.
4. **Exaggeration or Minimization:** Trying to make things larger or smaller by repeating them excessively.
5. **Doubt:** Bringing someone's or something's trustworthiness into question.
6. **Appeal to Fear or Prejudice:** Attempting to gain support for an idea by stirring up fear and panic among the population about an alternative, possibly based on prejudices.
7. **Flag-waving:** Exploiting strong national sentiments or relating to a population group, like race, gender, or political preference, to justify or promote an initiative or concept.
8. **Causal Oversimplification:** Assuming a single cause when several causes are related to a problem. It also includes scapegoating: blaming one person or a group without considering the complexity of a problem.
9. **Slogans:** A short statement that can contain labeling and stereotyping.
10. **Appeal to Authority:** Assertion that a claim is true just because a valid authority or expert supports it, without further evidence.
11. **Black-and-white Fallacy or Dictatorship:** Pretending that two choices are the only options when there are more.

12. **Thought-terminating Cliche:** Words or phrases that prevent critical thought and meaningful debate on a particular topic.
13. **Whataboutism:** Discredit the other party's position by accusing them of hypocrisy while not directly disproving their arguments.
14. **Reductio ad Hitlerum:** Convincing an audience to dislike an action or idea by implying the idea is popular with groups that the target audience despises.
15. **Red Herring:** Bringing irrelevant material into the discussion such that Attention is distracted from the actual issues.
16. **Bandwagon:** Trying to convince the audience to participate and follow the course because "everyone else is doing the same thing".
17. **Obfuscation, Intentional Vagueness, Confusion:** Intentionally using vague language to let the audience make their interpretation.
18. **Straw Man:** When the opponent's claim is substituted by a related one but then refuted in place of the initial one.

Given the Propaganda Technique Corpus with the different propaganda techniques, the dataset serves as the foundation of the Propaganda Detection Model and the subsequent analysis.

2.2 What is Deep Learning?

Deep Learning, a significant subset of Machine Learning, has revolutionized various fields by enabling machines to process images, music, speech, audio, and sequential data like text. At its core, Deep Learning refers to the specific architecture of a Neural Network-based Learning System, which utilizes computational models with multiple Processing Layers to process raw input data efficiently. These layers handle different aspects of the input data and contribute to the system's overall performance [LBH15].

Before the emergence of Deep Learning, conventional Machine Learning techniques relied heavily on expertly designed Feature Extractors, which required extensive domain knowledge of a dataset paired with expertise in Machine Learning. This process involved crafting a Feature Vector, which was then input into a simple classifier to detect patterns in the data. The engineering of this Feature Vector was a labor-intensive task, eventually streamlined by Representation Learning. Representation Learning is a set of methods allowing a machine to automatically discover the necessary representations for Classification or detection tasks. Consequently, the previously hand-crafted Feature Vector is generated automatically, eliminating the need for deep domain expertise in the input data. This ability to learn representations is crucial in differentiating Deep Learning from conventional Machine Learning techniques [LBH15].

The Multilayer Architecture of Deep Learning systems is one of its distinguishing features. In a Deep Learning environment, Learning occurs chained, where raw input data is fed into the first layer, creating a more abstract representation of the input by applying simple yet non-linear methods. The representations become increasingly abstract as the data passes through each layer, enabling the machine to use highly complex functions to extract knowledge from the data. Upon closer examination of the layers, their complexity is manageable: They comprise simple modules capable of solving complex problems when interconnected [LBH15].

During Deep Learning, the goal is to generate representations of data with multiple levels of abstraction. These representations can be achieved by discovering intricate structures in large data sets using the Backpropagation algorithm. This algorithm indicates how the Learning System should change its internal parameters to optimize the representations in the different Processing Layers. It is important to note that, in this process, each layer learns from the previous layer, so Learning occurs chained. At its core, the Backpropagation procedure is the practical application of the chain rule for derivatives. This rule states that the gradient of an objective concerning the module input can be computed by working backward from the gradient concerning the module output. The Backpropagation equation can be repeatedly applied to propagate gradients through all modules, starting from the prediction output and returning to the module where the raw input data is introduced [LBH15].

2.3 Various Concepts in Deep Learning

2.3.1 Preprocessing

Deep Learning models need a specific representation of input data to consume and learn from the data. Whenever the original format of the raw data is not usable by a model, Preprocessing techniques are applied [GBC16].

Stopwords

Stopwords are common words, such as *and*, *are*, and *this*, that appear frequently in documents but do not contribute to the sentence context. Their high frequency of occurrence can hinder a model from understanding the content. Thus those words can be removed during Preprocessing [KGV+14].

Stemming

Stemming is a text processing technique used in Information Retrieval, which reduces variant forms of a word to their standard stem. For example, *presentation*, *presented*, and *presenting* are stemmed to *present*. It operates on the assumption that searching for one form of a word implies an interest in all its variants [KGV+14].

Tokenization

Tokenization converts a text block, such as words or phrases, into smaller units called tokens. The process helps identify meaningful keywords, ensuring document consistency by addressing punctuation marks, different number and time formats, and standardizing abbreviations and acronyms [KGV+14].

Embedding Generation

When working with Transformer-based Language Models, it is crucial to perform Tokenization to generate a list of subwords. Each subword is then mapped to a unique integer from the model's vocabulary. In the next step, the integer vector is converted into a vector representation used as an input for the Transformer model [VSP+17]. The mapping is different with every Transformer-based Language Model since every model has its unique vocabulary.

2.3.2 Hyperparameter Optimization

Building a Deep Learning model requires different architectural decisions, like optimizing Hyperparameters, the number of layers, filters per layer, and the type of Activation Function. Initial choices may only sometimes be optimal due to the absence of formal rules [Cho17].

Automatic Hyperparameter Optimization has been developed to efficiently explore possible decisions, reducing the need for time-consuming manual adjustments. Optimization involves automatically selecting Hyperparameters, building and training the model, evaluating performance on validation data, and then iterating this process with a new set of Hyperparameters. The new set of Hyperparameters is determined based on past validation performance. The selection is crucial and can be done by employing various techniques, like Bayesian Optimization or Random Search [Cho17].

Contrary to training model weights, updating Hyperparameters requires creating and training a new model, as Hyperparameters are typically discrete and non-differentiable. Random Search is commonly used for Hyperparameters Optimization, but different Python libraries provide more efficient solutions [Cho17].

However, a significant concern is Overfitting to the validation dataset as Hyperparameters are updated based on validation data. Ultimately, Hyperparameter Optimization is essential for achieving top-performing models or succeeding in Machine Learning competitions [Cho17].

2.3.3 Data Augmentation

Deep Learning requires a large amount of labeled data to work well. During Supervised Data Augmentation, a sentence is modified to resemble the original example. This way, the augmented example can still be labeled under the same category as the original sentence. The synthetic data is then used with the original data to train a Classification model [XDH$^+$20].

Xie et al. [XDH$^+$20] proposed a different method, termed Unsupervised Data Augmentation. The procedure can be summarized by first computing an output distribution from an input sentence. In parallel, a noised version is created by injecting noise, and the divergence between the two versions is minimized to calculate a final loss. This way, the model gets insensitive to changes in the input space. Simultaneously minimizing the consistency loss gradually propagates label information from labeled examples to unlabeled ones.

Another possible Advanced Data Augmentation strategy for Text Classification is Backtranslation. A sentence is translated into a target language and then returned to the original language. This way, some words or even the sentence structure may change, but the semantics of the original sentence are preserved. Another possible strategy for Text Classification is word replacing with Term Frequency Inverse Document Frequency (TF-IDF). This strategy is helpful if specific keywords with high importance must be preserved in the augmented variation. Words with lower importance, thus with low TF-IDF scores, can be replaced with other low-score words [XDH$^+$20]. Another possible approach is Synonym Replacement [DFW20].

2.3.4 Inference

Inference refers to the process where relationships between variables are deduced. Unknown or hidden variables can be predicted by examining known values or observed variables. This process is often vital for performing other tasks or implementing learning rules. These learning rules are typically founded on the Principle of Maximum Likelihood, which aids in determining the most probable outcomes. Inference, therefore, involves predicting values or probabilities of unknown variables, given the known values of other variables [GBC16].

2.3.5 Postprocessing

After a Deep Learning model has produced its predictions, these results can be adjusted during Postprocessing. This stage can involve pruning, filtering, or integrating additional knowledge [Bru01].

2.3.6 Generalization, Over- and Underfitting

Generalization describes performing well on new, unseen inputs, not just those used during training. The training dataset is used to compute and minimize the training error. Generalization also aims to minimize the validation error, the expected error on a new input [GBC16].

The error is typically estimated by measuring the model's performance on a separate validation dataset. In the Machine Learning process, parameters are not predetermined.

Instead, the training dataset is used to adjust the parameters to lower the training error, and then the validation dataset is evaluated. As a result, the expected validation error is usually equal to or higher than the training error [GBC16].

A Machine Learning algorithm's success is determined by its ability to reduce the training error and the gap between the training and validation error. These are directly related to the key challenges in Machine Learning: Underfitting, where the model cannot achieve a low enough training error, and Overfitting, where the gap between training and validation error is too high. These challenges represent the factors that Machine Learning models must balance to be effective [GBC16].

2.3.7 Cross-Validation

The network evaluation can be achieved by adjusting various parameters, such as the number of epochs used for training, by dividing data into training and validation datasets. When dealing with a smaller dataset, the validation dataset could be limited, perhaps around 100 examples, leading to a potentially high variance in validation scores. This significant fluctuation in validation scores, dependent on which data points are selected for validation and training, can hinder a reliable model evaluation. In such instances, it is recommended to employ K-fold Cross-Validation. This method involves dividing the data into K partitions creating K identical models, and training each one on K–1 partitions while evaluating the remaining partition. The model's validation score is then determined as the average of the K different validation scores obtained, offering a comprehensive evaluation of the model's performance across different data partitions [Cho17].

2.3.8 Ensembling

Ensembling is a technique that involves training multiple different models on the same data and combining their predictions to make a final prediction. The main goal of Ensembling is to produce a final model that is more accurate and robust than any of the individual models [ZWT02].

2.3.9 Meta Classification - Stacked Generalization

Stacked Generalization is a form of Meta Classification. It is introduced as a methodology to reduce the Generalization Error of one or multiple models. The concept is based on determining the bias of the models relative to a given learning set. This discernment is achieved through a second round of Generalization, which operates in a distinct space. In this secondary space, the inputs are typically the predictions of the original models, which have been trained on a portion of the learning dataset and attempt to predict the remainder. The output of the second space is typically the correct prediction [Wol92].

2.4 Deep Learning Architectures

2.4.1 Feedforward Neural Network

A typical architecture in Deep Learning is a Feedforward Neural Network. These networks learn to map a fixed-size input to a fixed-size output, for example, a probability (output) for a category (input). To traverse the layer structure, a set of units compute a weighted sum of their inputs from the previous layer and pass the result through a non-linear function. The units between the Input and Output Layers are called Hidden Layers. They distort the input data non-linearly so that the different categories become linearly separable by the Output Layer [LBH15]. In Deep Learning, non-linear functions are often called Activation Functions.

Different kinds of Output Layers exist, like the Linear Layer or the Softmax Layer. A Linear Layer is a relatively simple example of an Output Layer, and it linearly splits the distribution into two distinct parts. The layer can only learn Linear Relations, making them useless for learning non-linearity. The Linear Layer reduces the previous layers' dimensions to ease data interpretation during Learning. It takes a flattened one-dimensional vector as input and is multiplied by a weight matrix, yielding the output feature [GBC16].

A Softmax Layer is used whenever a Probability Distribution is needed. Most often, Softmax Layers is used as the output of a classifier. However, they can also be used inside the model itself if the model has to choose between multiple options between some internal variables. When applying Softmax for Multi-class Classification, each output probability for the distinct classes has to lie between 0 and 1. The different output probabilities are outputted as a vector, with the specialty that all probabilities sum up to 1. The Softmax Function outputs zero for a class if it clearly cannot be classified as such, while one corresponds to absolute certainty [GBC16]. Simple Feedforward Neural Networks are often also called Multilayer Perceptrons.

2.4.2 Convolutional Neural Networks

A Convolutional Neural Network is a specific type of Feedforward Network that can generalize much better than networks with full connectivity between adjacent layers. Convolutional Neural Networks process data structured in multiple arrays. The four critical concepts behind Convolutional Neural Networks include Local Connections, Shared Weights, Pooling, and the Inclusion of Multiple Layers. The typical architecture consists of a series of stages, with the first stages typically being Convolutional and Pooling Layers. The Convolutional Layer has units organized in Feature Maps, and each unit connects to its previous layer through local patches within the Feature Map. This connection is established through a set of weights. While the primary role of the Convolutional Layer is to detect local conjunctions of features from the previous layer, the Pooling Layer merges semantically similar features. These merges can be done by computing the maximum of a local patch of units in one or a few Feature Maps. Another alternative is Neighboring Pooling, where the input is shifted by more than one row or column. Convolution Stages typically consist of two to three components: Stacked Non-Linearity and Pooling and Convolutional and Fully-connected Layers. Finally, Backpropagation optimizes the parameters for training. Convolutional Neural Networks are particularly well-suited for Image Recognition and processing tasks due to their ability to capture spatial information and hierarchical patterns in data [LBH15].

2.4.3 Recurrent Neural Networks

Recurrent Neural Networks were one of the main architectures that benefited from the introduction of Backpropagation. This type of network is used for tasks that involve sequential inputs, such as speech and language. Recurrent Neural Networks process only one input sequence at a time. In their Hidden Layers, Recurrent Neural Networks contain a State Vector, which preserves information about past elements of the sequence. This recurrent connection enables the network to maintain a memory of previous inputs, allowing it to learn and model temporal dependencies in the data. All layers share the same weights, supporting their primary purpose in learning long-term dependencies. However, it is difficult for Recurrent Neural Networks to store information for an extended period. This limitation can be addressed by incorporating explicit memory components [LBH15].

2.4.4 Long Short-Term Memory Network

One approach to incorporating explicit memory is the Long Short-Term Memory Network (LSTM). The LSTM architecture uses specialized Hidden Layers, which enable a Recurrent Neural Network to remember inputs for a long time. A memory cell acts like an accumulator, connecting the current input sequence to the subsequent one. The memory cell maintains its real-valued state and accumulates external signals. The Self-Connection is gated by another unit that learns when to clear the memory content. The LSTM is an improved version of the Recurrent Neural Network, which tackles the problems of exploding and vanishing gradients during Backpropagation [LBH15]. A drawback of LSTMs is their lacking knowledge of future elements of the input sequence. Bidirectional LSTMs was introduced by Schuster and Paliwal [SP97] to address this problem. These networks consist of two separate LSTMs, one processing the input sequence in the forward direction, the other in the backward direction. The Hidden States of both LSTMs are combined at each computation step to make predictions. This way, both past and future context is captured.

2.5 Transformers in Deep Learning

2.5.1 The first Transformer Model

Introduced by Vaswani et al. [VSP$^+$17] in 2017, the Transformer architecture revolutionized Sequence Transduction models. Before this, prevailing models relied on intricate

Recurrent or Convolutional Neural Networks combined with an Encoder and Decoder. As the most effective models featured an Attention Mechanism linking the Encoder and Decoder, the authors presented a novel network architecture centered exclusively on the Attention Mechanism, making complex Recurrent and Convolutional Networks obsolete. By minimizing sequential computation, Transformers can better comprehend dependencies between distant positions, albeit at the expense of reduced effective resolution. This issue is mitigated through Multi-Head Attention.

The original Transformer model employs the Encoder-Decoder Structure, integrating Stacked Self-Attention and a Fully-connected Layer for both Encoders and Decoders. The Encoder within the Transformer architecture converts an input sequence of symbol representations into a series of continuous representations. Subsequently, the Decoder processes the continuous representation one element at a time, generating an output sequence [VSP+17]. Each representation is created incrementally, reflecting the auto-regressive property of Transformer models. Every new output sequence utilizes the latent representation of the previously formed sequence during generation [GCMK20]

2.5.2 Attention in Transformers

The Attention function in Transformers is a robust process that maps a query and a set of key-value pairs into an output, using vectors to represent the information. This output is generated as a weighted sum of the values, where a Compatibility Function between the query and the respective key in each pair determines the weights. Vaswani et al. [VSP+17] introduced the Scaled Dot-Product Attention, which involves calculating and scaling the Dot Product of queries and keys before applying a Softmax Function to obtain the weights for the values. These weights and the key-value pairs' values are then utilized for another Dot Product computation.

The Multi-Head Attention concept was developed to enhance learning capabilities, enabling multiple Attention functions to operate simultaneously. The enhancement is achieved by projecting the queries, keys, and values linearly with previously learned projections and running the Attention function in parallel on all projected versions of the queries. The resulting outputs are combined and projected to create the final output. This method allows the model to learn from different representation subspaces at various positions. In Transformer models, Attention operates in three distinct positions: within the Encoder and Decoder structure, inside the Encoder itself, and within the Decoder [VSP+17].

The advantages of Self-Attention lie in its superior parameter efficiency and adaptability when dealing with inputs of varying lengths, designed to model long-range dependencies using a fixed number of layers. Another key advantage of Self-Attention is its parallelization ability due to its steady sequential operations and maximum path length, which matches the length of Fully-connected Layers [LWLQ22].

2.5.3 Encoder and Decoder Structure in Transformers

Encoder and Decoder each use six identical layers, whereas the Encoder has two additional sublayers with every layer. In contrast, the Decoder has a third additional sublayer. The first two sublayers of both are the same, where the first is a Multi-Head Self-Attention mechanism, end the second is a simple, position-wise, Fully-connected Feedforward Network. The third sublayer of the Decoder uses the stacked Encoder output and performs Multi-Head Attention. A Residual Connection, which connects an output of a layer to the input of another subsequent layer, and Layer Normalization, meaning all neurons in a particular layer effectively have the same distribution across all features for a given input, are employed on all of these sublayers. To ensure the predictions can only be made on previously encountered outputs, the Self-Attention sublayer during decoding is modified to prevent positions from attending to subsequent positions. Since the Decoder can only work with previous representations, the first Transformer is a unidirectional network [VSP+17]. The Encoder itself is bidirectional, while the Decoder is unidirectional [GCMK20].

There are three ways to incorporate the Encoder and Decoder structure in Transformers. Combining Encoder and Decoder is helpful for Sequence-to-Sequence Modeling, like in Neural Machine Translation. An Encoder-only Architecture is often used for Natural Language Understanding. In this case, the outputs of the Encoder serve as a representation of the input sequence. Finally, the Encoder and the Encoder-Decoder Cross-Attention modules are removed in a Decoder-only architecture. This removal makes a Transformer suitable for Sequence Generation like Language Modeling [LWLQ22].

2.5.4 The Evolution of Transformer Models

As Transformers emerged as the top choice in Natural Language Processing, numerous modifications and variations were introduced to enhance their architecture and design. These improvements addressed a range of issues, such as the efficiency of handling long sequences, which were previously impaired by the computational and memory demands of the Self-Attention module. In addition, Generalization was boosted, overcoming difficulties in training with limited data, as the original Transformer architecture lacked assumptions on the structural bias of input data. Later iterations of Transformers were tailored to various downstream tasks and applications, making them versatile in fields like Natural Language Processing, Computer Vision, and Speech Processing [LWLQ22]. For Propaganda Detection, Transformers' Natural Language Processing capabilities are most valuable.

2.5.5 Overview of Transformers-based Language Models

Transformers have advanced significantly, demonstrating superiority over traditional Recurrent and Convolutional Neural Networks. Early Transformer models such as Generative Pre-trained Transformer (GPT) [RNSS18] and Generative Pre-trained Transformer 2 (GPT-2) [RWC+19] were limited to generating outputs based on prior context due to their unidirectional Decoder within the Encoder-Decoder structure [VSP+17]. To overcome this limitation, bidirectional networks emerged, discarding the unidirectional Decoder and focusing on a bidirectional Encoder. Bidirectional Encoder Representations from Transformers (BERT) [DCLT19] learns from a sentence's past and future context. XLNet [YDY+19] adopted BERT 's Auto-Regressive Network Architecture and bidirectional Encoder. Zhuang et al. [ZWYJ21] proposed three enhancements to BERT, resulting in Robustly Optimized BERT Pre-training Approach (RoBERTa), which was trained on a larger dataset with increased batch size and extended pretraining. However, the larger model required substantial resources for training. Subsequent models, such as ALBERT [LCG+19] and DistilBERT [SDCW19], reduced model size by factorizing Embeddings and implementing Cross-layer Sharing or using Knowledge Distillation, respectively, to create smaller yet powerful networks [GCMK20].

2.5.6 Historical Important Models

The chosen historical models marked the beginning and first advances in Transformer-based Language Models.

BERT

BERT is a Transformer-based Language Model released by Devlin et al. [DCLT19]. Besides previous architectures, BERT is designed to pretrain deep bidirectional representations from the unlabeled text by utilizing all layers' left and right input context. Pretraining is done only once during model creation. The pretrained model can be customized for various tasks by adding one fine-tuned Output Layer. A labeled dataset is used to fine-tune the pretrained BERT model for a specific downstream task during Fine-tuning. Even though the downstream tasks may differ, the base, such parameters, and data used are always the same [DCLT19]. The input representation can represent a single sentence and a pair of sentences in one input sequence. The vocabulary of BERT spans 30000 tokens. The first token of the input sequence is a unique Classification Token, which contains the whole aggregated representation of the sequence from the previous and actual layers. This special token is refered as *[CLS]* in BERT. Another special token used to separate the two distinct input sentences in pair of sentences scenario is the Separator Token: *[SEP]*. Further, a marker is added to every token to indicate belonging to the first or second sentence. An input representation is constructed by summing a Token-, Segment-, and Position Embedding for every token. During Pretraining, BERT uses Masked Language Modeling and Next Sentence Prediction. In Masked Language Modeling, a random sample of tokens on the input sequence is replaced with another special token: The Mask Token *[MASK]*. From the input tokens, 15% are selected for possible replacement, of which 80% are replaced, 10% are not changed, and a random vocabulary token replaces the left-over 10%. The Next Sentence Prediction predicts whether two segments follow each other in the original text. Its main objective is to improve performance on downstream tasks, which require knowledge about the relationship between pairs of sentences [DCLT19].

RoBERTa

After the success of BERT, a robustly optimized version was created by Zhuang et al. [ZWYJ21]: RoBERTa. RoBERTa is an extension of BERT with changes to its Pretraining phase. The model was trained longer, with bigger batches and more data. The Sentence Prediction module was removed entirely, and the sequence length was enlarged to a maximum of 512 tokens. No randomly short sequences were injected during training, and the model was trained with full-length sequences. During training, Dynamic Masking was introduced. BERT was pretrained with Static Masking: A mask was generated during Preprocessing and used for every training epoch. Zhuang et al. [ZWYJ21] changed this by generating a mask every time a new sequence is fed to the model. Due to these changes, the newly created RoBERTa model outperforms BERT in various tasks [ZWYJ21].

Cross Language Models (XLMs)

Low-resource languages typically need more substantial labeled and unlabeled data. By exploiting the capabilities of Multi-lingual Models during the Fine-tuning process, labeled data from multiple languages can be utilized to enhance performance in downstream tasks. Conneau et al. [CKG+20] developed Cross Language Model RoBERTa (XLM-R), a RoBERTa-based Cross-Language Model. Unlike other Multi-lingual Models, XLM-Rs benefits from more extensive training data and covers a larger range of languages, including those with limited resources.

2.5.7 Advanced Transformer-based Language Models

Since much progress was made in Artificial Intelligence and Transformer-based Language Models, this chapter will look at the current prevailing models to give an overview of the current state of the art. While the list of models is growing at scale, the below-mentioned models are by no means meant to be complete. The models were chosen due to public interest, like those developed by OpenAI[1], or by their direct relevance to the research community, like the open-sourced models by Meta[2].

Generative Pre-trained Transformer 3 (GPT-3)

2020, GPT-3, an Auto-Regressive Language Model, stands out with its remarkable configuration of 175 billion parameters - a tenfold increase compared to the previous Language Models. Its performance was examined in a Few-shot Setting, absent any need for gradient updates or Fine-tuning. All tasks and Few-shot demonstrations were conveyed to GPT-3 solely through textual interaction. The model demonstrated its capabilities across various datasets, covering translation and question-answering. Furthermore, it successfully handled tasks requiring on-the-fly reasoning or domain adaptation, like word unscrambling, use of an unfamiliar word in a sentence, or carrying out 3-digit arithmetic.

[1] https://openai.com (last accessed: 10 May 2023)
[2] https://ai.facebook.com (last accessed: 10 May 2023)

GPT-3 can generate news article samples that human evaluators struggle to differentiate from human-written pieces [BMR+20].

Generative Pre-trained Transformer 4 (GPT-4)

In 2023, OpenAI released GPT-4, a novel Multi-modal Model that consumes image and text inputs to yield text outputs. A central goal of this model is to boost its ability to interpret and generate natural language text, more so in scenarios requiring greater complexity and nuance. Evaluations designed for humans were employed to assess GPT-4's capabilities and manifested exceptional performance, often surpassing many human test-takers. For instance, on a mock bar exam, GPT-4's score featured in the top 10% bracket, contrasting with GPT-3, which languished in the lowest 10% [Ope23]. The bar exam is a professional test that law school graduates must pass to practice law in a specific jurisdiction or state in the USA.

The GPT-4 developers emphasized the formulation of a Deep Learning stack that scaled predictably - a necessity given the nonviable nature of comprehensive model-specific tuning for substantial training runs. The research team addressed this by innovating infrastructure and optimization strategies that displayed uniform behavior across varied scales. These advancements facilitated the reliable prediction of some facets of GPT-4's performance using smaller models trained with computational resources ranging from 1,000 to 10,000 times less [Ope23].

Open Pre-trained Transformer (OPT)

OPT was developed with a clear objective - to promote reproducible and responsible research at scale and to facilitate broader participation in the discourse on the impacts of these Large Language Models (LLMs). It is vital to have a community-wide understanding of risk, harm, bias, and toxicity in LLMs, which can only be achieved when these models are accessible for scientific research [ZRG+22].

LLMs, often trained over hundreds of thousands of computing days, exhibit extraordinary Zero- and Few-shot Learning skills. However, given their considerable computational cost, these models remain challenging to reproduce without significant financial resources. Furthermore, in the few instances where these models are accessible via Application Programming Interfaces, access to the entire model weights is typically denied, complicating their analysis. This restricted access has curtailed researchers' ability to explore how and why these LLMs function, thereby impeding their comprehension [ZRG+22].

A suite of Decoder-only pretrained Transformers, OPT, has been developed in response to this issue. These Transformers, ranging from 125 million to 175 billion parameters, are fully and responsibly shared with researchers. OPT models have been designed to align with the performance and sizes of the GPT-3 class of models, incorporating the latest best practices in Data Collection and efficient training [ZRG+22]. It has been demonstrated that OPT-175B is on par with GPT-3. Nevertheless, it requires only a seventh of the carbon footprint to develop, demonstrating its efficiency and sustainability [ZRG+22].

Large Language Model Meta AI (LLaMA)

The previous models were trained based on the hypothesis that increased parameters will correspondingly enhance performance. However, recent research has shown that the largest models may not perform best, but smaller models trained on more data [HBM+22]. LLaMA is a sequence of LLMs optimized for optimal performance across varying Inference budgets by training on more tokens than is typically used. The developed models range from 7 billion to 65 billion parameters and demonstrate competitive performance compared to other leading LLMs. Notably, despite being ten times smaller, LLaMA-13B surpasses GPT-3 in performance on most benchmarks. The model can run on a single Graphics Processing Unit (GPU) and helps researchers explore the capabilities, biases, and limitations of such Large Language Models. At the higher end of the scale, the LLaMA-65B model remains competitive with the top-tier LLMs [TLI+23].

2.6 Generating Custom Embeddings

The chapter gives an overview over different strategies that can be used to enrich Transformer-based Language Model Embeddings.

2.6.1 Embeddings from Language Models (ELMo)

ELMo offers a dynamic approach to word representation, capturing the complex aspects of word usage, such as syntax and semantics, and embracing variations in linguistic contexts, essentially addressing multiple possible meanings for a word. By utilizing bidirectional Deep Learning Models, word vectors are derived from the internal states of these models, pretrained on extensive text corpora to ensure richness in capturing the essence of words [PNI+18].

When a word vector spans a whole sentence, it is called a Sentence Embedding. In Transformer-based Language Models, where multiple Hidden Layers are employed, the output of the last Hidden Layer is often called the Aggregated Sentence Embedding for a Classification Task [DCLT19]. Another approach is to take the Embeddings from all Hidden Layers and average the values to gain a single Averaged Embedding.

2.6.2 Bag-of-Words

The Bag-of-Words technique provides a simple yet effective way to analyze text. It collects words from a text into a set, or *bag*, paying attention to their frequency but not their order or sentence structure. This process often involves removing stopwords. The advantage of Bag-of-Words is its simplicity and no need for linguistic knowledge. However, its limitation lies in its inability to capture more complex aspects like sentence structure or semantic context [MCG16].

2.6.3 TF-IDF

A standard method used in Text Classification is the TF-IDF approach. TF-IDF assigns a weight to each word in a document based on its uniqueness, effectively capturing the relevance of words within the text sequence and its corresponding category. This relevance means that words frequently used in a specific text sequence but less common in other sequences will have a higher TF-IDF score, indicating their importance and relevance to that particular sequence [YtLYc05].

2.6.4 Part-of-Speech Tags

Part-of-Speech Tagging assigns each word in a given text a label that denotes its grammatical role, whether a noun, verb, adverb, or preposition. These tags can be beneficial in specifying particular items in regular expressions, such as proper nouns for names. They can also assist in clarifying words with multiple potential tags. For instance, *book* can be a noun or a verb. However, in the sentence *book a flight*, *book* should be tagged as a verb because a noun would not be grammatically correct before a determiner and another noun [MCG16].

2.6.5 Named Entity Recognition

Named Entity Recognition is a subtask of Information Extraction that identifies entities such as individuals, organizations, dates, and locations within a text. Named Entity Recognition aims to extract specific information from unstructured text and convert it into a structured, machine-readable format. This process is crucial for building summaries, knowledge bases, and ontologies, providing a structured representation of the data [MCG16].

2.7 Deep Learning Strategies

This chapter gives an overview of different Deep Learning terms.

2.7.1 Supervised Learning

A common form of Deep Learning is Supervised Learning. During the first step, a large dataset is gathered. This dataset must represent a real-world case from which the machine can learn. The input data is then categorized into different categories called labels. Those labels form the desired output made by the Deep Learning system. The Deep Learning system is shown the input data and its label during learning. The machine must then conclude how the input data can be used to predict the label. The machine generates and refines its outputted Feature Vector with every learning step. In an ideal setting, the final refined output vector would look precisely like the vector representing the to-be-predicted label. In this case, the machine is sure to predict the desired category. The Deep Learning system calculates a score for every category available. The category with the highest score is then predicted by the algorithm as its supposed guess. The score is calculated by a function that measures the difference between the predicted and actual labels. The distance can be calculated since both are represented in abstract vectors. Once the difference between the two vectors has been calculated, the Deep Learning algorithm changes its internal parameter to reduce the error. Those parameters are called weights, and hundreds of millions may be in a single Deep Learning system [LBH15].

The weights are adjusted by calculating a Gradient Vector. This vector shows how a slight change in the weight would change the difference between the true and the predicted value. The overall goal of a Deep Learning system is to reduce the error until no further optimization is possible and to make the error as small as possible. A strategy called Stochastic Gradient Descent is utilized to tackle the Minimization Problem in practice. During Stochastic Gradient Descent, the input vector is checked on a few examples. Then, an output is generated with the resulting error terms. The Average Gradient is calculated for those few examples, and the weights are adjusted accordingly. This calculation is done as long as the average of the objective function stops decreasing. Finally, after the model has been trained, a separate part of the training data is used to test the model's ability to predict the correct label on unseen data [LBH15].

2.7.2 Semi-Supervised Learning

Unfortunately, it is often difficult or expensive to collect and label training data for Supervised Learning [WKW16]. Semi-Supervised Learning combines Supervised and Unsupervised Learning [CCZ06] [Zhu08]. Often Semi-Supervised Learning is utilized to improve a model in either a Supervised or Unsupervised Learning environment. Solving a Classification Problem in Supervised Learning might require additional data points to improve the classifier's output. Semi-Supervised Learning has also been applied to areas where labeled data exist. However, the unlabeled data brings new helpful information for prediction so that the performance can be improved [vEH20].

2.7.3 Unsupervised Learning

During Supervised Learning, a set of data points consisting of some input data and its corresponding output value is provided. In Unsupervised Learning, no specific output value is provided. Instead, the model tries to find an underlying structure from the input data [WKW16]. Unlabeled data is used to train a Deep Learning Algorithm based on its encountered features from the dataset.

2.7.4 Self Training

Self Training uses a single supervised classifier that has been iterative trained on labeled data and pseudo-labeled data [vEH20]. It is called pseudo-labeled data since an insufficient model categorizes unlabeled data. The most confident predictions are added to the original data, and the supervised classifier is trained on the new and original data from before until no more unlabeled data remains. The procedure of Self Training requires different design decisions, such as the selection of data to be pseudo-labeled and the re-usage of this data for learning [vEH20]. The pseudo-labeled data is sometimes referred to as *Silver Data*.

2.7.5 Minority Classifier

A Minority Classifier refers to a Machine Learning model trained to identify and categorize instances of the Minority Class in an imbalanced dataset. Imbalanced data is a common problem in Machine Learning, where unequal classes exist. The Minority Class has fewer instances and is often the more important class in problems such as Fraud Detection or Disease Diagnosis [NS16].

2.7.6 Hierarchical Classification

Hierarchical Classification is a Classification Problem where the classes are organized into a hierarchical structure. Unlike other Classifications, where each class is treated independently, Hierarchical Classification acknowledges and utilizes the relationship between classes. This type of Classification is beneficial in scenarios where classes naturally form a hierarchy [SF11].

2.7.7 One-Versus-One Classifier

A One-vs-One Classifier is a method for creating an individual classifier for each category pair. If there are three classes (Class 1, Class 2, Class 3), a classifier for Class 1 vs. Class 2, another for Class 1 vs. Class 3, and finally, one for Class 2 vs. Class 3 is created. When determining the class of a new item, the item is run through all the classifiers. Each classifier votes on what they perceive the item to be, and the class that accumulates the most votes is the winner. An excessive number of classes can result in a large number of classifiers. However, the method often delivers more accurate results than other methods. The alternative is a One-versus-All classifier, where only one classifier is trained, that calculates the probabilities for one item being part of the given classes [MG13].

2.7.8 Single- vs. Multi-class Classification

Single-class Classification is a Machine Learning problem where each example is categorized into one of two classes. On the other hand, Multi-class Classification is when each item can belong to multiple predefined categories. Both types of classification aim to build a learning model from labeled training data that can accurately predict the category of new, unlabeled objects [MG13]. Deriving from this, in a Multi-label Multi-class Classification scenario, each example is not restricted to a single label but can be associated with several labels out of many potential ones. Simply put, classes in this context are not exclusive, and an example can be classified under multiple categories simultaneously.

2.7.9 Transfer Learning

High-performing models can be created by utilizing data from different domains. This technique is called Transfer Learning, where information from one domain can improve a learner from a related but different domain. Many domains seem distinct, but when looking at their high-level domain, they often share specific characteristics [WKW16].

2.8 Additional Topics

This chapter explains further relevant topics in the context of the thesis.

2.8.1 Ridge Regression

In ordinary Linear Regression, a line is sought that minimizes the sum of the squared differences between the actual and predicted values. However, when predictor variables are highly correlated, this can lead to problems. The model becomes unstable, meaning small changes in the data can lead to significant changes in the predictions. Also, the model can overfit, meaning it is too closely tailored to the training data and needs to perform better on new data. Ridge Regression addresses this by adding a penalty term to the equation that the model is trying to minimize [HK00].

2.8.2 Longest Common Subsequences

The Longest Common Subsequence is a concept used to compare two strings. It refers to the maximum number of identical symbols or elements found in both strings while maintaining the order of these symbols. This similarity measure is vital in many fields, including spell-checking applications, molecular biology, and file archiving systems. In each of these cases, the aim is to assess how alike two sequences are. When comparing words, the Longest Common Subsequence provides an effective metric for evaluating the degree of resemblance. The larger the Longest Common Subsequence, the closer the two strings are considered to be [BHR00].

2.8.3 Conditional Random Field

Conditional Random Field (CRF) is a technique for segmenting and labeling sequential data. First, an undirected graphical model is employed. A single Log-linear Distribution is defined over sequences of labels given a particular sequence of observations. The unique advantage of CRFs lies in their conditional nature. A label is not predicted based solely on an individual sample, but the context of surrounding samples is also considered. Dependencies between samples can be modeled as individual predictions are incorporated into the graphical model. This results in a powerful and flexible tool for dealing with sequential data [LMP01].

2.8.4 Evaluation Metrics

The performance of a Deep Learning model can be measured with the help of the most common Evaluation Metrics in the Machine Learning domain. These are Accuracy, Precision, Recall, and the F1-Score. The Recall is the True Positives Rate, which tells how many propaganda techniques were predicted correctly. Further, Recall tells us how many of the Real Positives are detected by a model. So, if it is known that a specific propaganda technique appears ten times, Recall checks this with the encountered number of the propaganda techniques. Precision denotes how many Predicted Positives are truly Real Positives. Precision tells how accurate a model is when comparing the predicted and actual propaganda techniques. Precision and Recall are insufficient to assess a model's quality in some situations. The F1-Score is an approach to balance Precision and Recall. The Harmonic Mean, named F1-Score, references the True Positives to the Arithmetic Mean of Predicted Positives and Real Positives [Pow11]. The focus during performance evaluation is on the Micro F1-Score since the challenge dataset has an uneven class distribution and a balance between Precision and Recall is needed. Also, the F1-Score was used during the challenge as the prioritized measure for assessing participant's performance [DSMBCW^{+}20].

CHAPTER 3

Literature Review

This chapter hosts the review of important literature, describes the search process and finally the analysis of all submissions of the SemEval 2020 Task 11 challenge.

3.1 Reviewing the Literature

The Literature Review is an essential part of the research process. It helps to find important academic sources that add depth to scientific work and makes it easier to understand the specialized terms and concepts used. It is also helpful in creating a list of relevant sources for the research topic. After searching for literature, the next step is to review the found resources and determine their relevance to the research [RS04].

3.2 Searching the Literature

The first step is to search Google Scholar[1] to find all the papers related to the *SemEval 2020 task 11* challenge. The search uses the specific phrase *intitle:"semeval 2020 task 11"*. This phrase helps filter out papers that include this exact phrase since all submissions for this challenge included *semeval 2020 task 11* in their title. The search results show 33 papers, which is close to the number of participants mentioned by Da San Martino et al. [DSMBCW+20].

The final step performed was a Backward Search to make sure no relevant publications had been missed by checking the mentioned participants by Da San Martino et al. [DSMBCW+20].

[1] https://scholar.google.com (last accessed: 22 December 2022)

3.3 Analysis of the SemEval-Challenge

After the challenge, 25 teams published a paper about their approach to detecting propaganda. The following chapter looks at every submission and categorizes the different approaches into categories: Preprocessing, Model Choice, System Setup, Postprocessing, and Ensembling Strategy. These categories will later be used to build mixed approaches from all the ideas the challenge participants incorporated. The approaches of the first five submissions will be analyzed in greater detail, while further submissions will be scanned for ideas and overall approaches.

3.3.1 Overall Analysis of Model Choice

The participants used different models, with a clear trend to using the Transformer architecture, more precise Transformer-based Language Models.

Jurkiewicz et al. [JBKG20], Chernyavskiy et al. [CIN20], Raj et al. [RJR$^+$20], Singh et al. [SSKM20], Grigorev and Ivanov [GI20] used only RoBERTa as their preferred Learning Model. Jurkiewicz et al. [JBKG20] do not state if they used a cased or uncased model, while Chernyavskiy et al. [CIN20] proposed the usage of a cased model.

Morio et al. [MMOM20] built an Ensemble Model consisting of BERT, RoBERTa, XLNet, XLM, Albert, and GPT-2, Kim and Bethard [KB20] also decided to use multiple models, namely BERT and RoBERTa.

Most of the challenge participants stick with BERT as their preferred Transformer model, like Dimov et al. [DKS20], Blaschke et al. [BKT20], Bairaktaris et al. [BSA20], Kaas et al. [KTP20], Krishnamurthy et al. [KGY20], Altiti et al. [AAO20], Jiang et al. [JGM20], Patil et al. [PSA20], Paraschiv and Cercel [PCD20], Li and Xiao [LX20], Daval-Frerot and Weis [DFW20], Dao et al. [DWZ20] and Kranzlein et al. [KBG20].

Additionally, Kaas et al. [KTP20], Patil et al. [PSA20], and Li and Xiao [LX20] stacked BERT with a Logistic Regressor, while Dao et al. [DWZ20] and Kranzlein et al. [KBG20] combined BERT with a LSTM layer.

Only four submissions decided to go without a Transformer architecture. Petee and Palmer [PP20] used a Logistic Regressor, Ermurachi and Gifu [EG20] tested a Random Forest approach, Arsenos and Siolas [AS20] still relied on Deep Learning, using a Multilayer Perceptron with a LSTM layer, as like Martinkovic et al. [MPS20] relying on LSTM in their submission.

3.3.2 Analysis of Best Five Submissions

Beginning with the best five submissions the different approaches in terms of Preprocessing, System Setup, Post Processing, and Ensemble are summarized.

Preprocessing

In Machine Learning, the input data is one of the keys to build an excellent predicting model. Therefore most of the challenge participants put a big emphasis on Preprocessing the given challenge dataset in different ways.

The first placed Jurkiewicz et al. [JBKG20] tested two approaches, of which the first was to use the left and right context of the propaganda span as the input, and the second to insert special tokens around the span, which should be classified as a propaganda class. The second approach leveraged the context again, this time without a special token around the propagandistic span. The context had the maximum possible size of 512 words. Therefore, the left context includes 256 subwords, like the right context.

The research by Chernyavskiy et al. [CIN20] reduces the Multi-label Multi-class Problem to a Single-Label Multi-class Problem by creating copies of spans with multiple labels. The construction of the input span involves a combination of the propaganda span with the corresponding sentence. This combination is divided by a Separator Token and initiated with a Classification Token.

A proposal in which no context was utilized was put forward by Morio et al. [MMOM20]. In their approach, the Sentence Embedding of the propaganda span was fused with Part-of-Speech Tags and TF-IDF Tags.

The study by Raj et al. [RJR+20] involved experimenting with three different ways of feeding data into their RoBERTa model. The initial method used only the propaganda span itself, while the second method used only the context of the span. Lastly, a combination of both the propaganda span and context was considered. The final chosen approach involved this combination of both elements.

Again, the span context was utilized in the research by Singh et al. [SSKM20]. Rather than merging the propaganda span with the surrounding context, these elements were individually introduced into two pretrained Transformers. The context was restricted to a total of 130 characters. The sequence representations from each model were retrieved and combined, resulting in a final Embedding that combines the contextualized outputs of both models.

System Setup

Three unique systems were utilized in the study by Jurkiewicz et al. [JBKG20]. The initial system made use of RoBERTa in both of its variants. The first variant ended at this stage, whereas the second added a small stacked Transformer on top of RoBERTa, which solely used the propaganda span's Embedding. This added Transformer was characterized by three Hidden Layers, four Attention Heads, and an Intermediate Layer of size 512. The second system was introduced in response to the significant imbalance in the dataset. Here, weights dependent on each class were utilized. These weights were computed by determining the Inverse Frequency with the frequency of the most prevalent class. Subsequently, these weights were incorporated into the loss function.

The final system, meanwhile, integrated the components of the first and second systems, complementing them with Self Training. No high-confidence examples were chosen, and no loss correction was done for noisy annotations. The top-performing model from the Span Identification task was repurposed to annotate 500,000 random sentences from OpenWebText[2]. As a result, the authors identified more sentences with propagandistic elements, which they then reused to annotate an additional dataset.

In their research, Chernyavskiy et al. [CIN20] developed two model versions. The first version used the input span, and the Sentence Embedding was obtained from the Classification Token. This token was passed through an additional Softmax Layer to generate a prediction. The second version of the model combined the extracted Classification Token from the Sentence Embedding with the Averaged Embeddings from the remaining propaganda span and the span length. Each Hidden Layer of a Transformer-based Language Model produces an Embedding for the propaganda span. As the process progresses, the sentence representation becomes increasingly accurate. An Averaged Embedding is created by calculating the mean of the Embeddings of all Hidden Layers for additional learning. A Fully-connected Layer was added on top of this. Furthermore, Transfer Learning was employed as a third strategy. The model was initially trained using data from the Span Identification subtask, and then further training was done during Technique Classification.

In the approach by Morio et al. [MMOM20], two separate Feedforward Networks are utilized, into which the concatenated input is introduced. The role of the first Feedforward-Network is to procure the sentence representation, whereas the second one is employed to achieve representation from all tokens within the propaganda span. The final model input is assembled by merging the sentence representation, the representations of tokens found at the start and end of the propaganda span, and finally, the representations garnered through Attention and Pooling. An additional label-wise Feedforward Network and a Linear Layer are incorporated to extract information specific to each propaganda technique. Furthermore, weights corresponding to the proportion of positive samples are allocated to the loss function to manage class imbalance. During Inference, labels predicted for each sentence are arranged in descending order and assigned to labels according to their order in a multi-label span.

In the study by Raj et al. [RJR$^+$20], three interconnected systems were constructed to classify the propaganda techniques. The first system adapts the RoBERTa model, modifying the final layer, resulting in 14 Hidden Layers, with the last being a Softmax Layer. This last layer is trained on the downstream Technique Classification task, while the other layers are fine-tuned beforehand. Due to the high imbalance in the dataset, a system termed the *Minority Classifier* is introduced. It comprises five separate Hierarchical Classifiers, termed as level-1 classifiers, focusing on the five Minority Classes. Each level-1 classifier is an Ensemble of 13 One-versus-One Classifiers, named level-2 classifiers. The outputs of these level-2 classifiers are collated to procure the prediction

[2] https://github.com/jcpeterson/openwebtext (last accessed: 23 December 2022)

for the level-1 classifiers. If the prediction confidence surpasses a certain threshold, the span is considered a positive example of the Minority Class. The level-2 classifiers are straightforward Linear Classifiers, aiding in faster computations during learning. The *Repetition* class was also managed separately, but not during Postprocessing like by Chernyavskiy et al. [CIN20] but during the training phase. This issue identifies the presence of the Longest Common Subsequences between the propaganda span and context. Rather than employing an exact match approach, they calculate the presence of *Repetition* by determining a percent match between the span and context, with a threshold that adjusts based on the length of the fragment.

After obtaining the concatenated Embedding, Singh et al. [SSKM20] passed it to a Classification Layer on top, which performs Technique Classification. However, before this, an additional Hidden Layer reduces the dimension of the Context Embedding. Since the reduced dimensions, the additional classifier gives more attention to the actual propaganda span.

Postprocessing

Chernyavskiy et al. [CIN20] experienced some difficulties with their first models, mainly when predicting the *Repetition* technique. This issue was addressed during the Postprocessing stage, where the presence of any given span was examined throughout the entire dataset. This process involved looking for exact matches; once punctuation was removed, stopwords were filtered out, and stemming was applied. The label for *Repetition* was assigned if the span in question matched at least two other spans. However, if only a single match existed, the classifier had to predict the label with a minimum threshold of 0.001. If no match was found, the probability was set to zero unless the classifier had predicted the label with a minimum probability of 0.99.

Ensemble

In their work, Jurkiewicz et al. [JBKG20] utilized an Ensemble method that averaged the class probabilities from their three developed systems, each trained with different Hyperparameters.

Similarly, Chernyavskiy et al. [CIN20] employed an Ensemble approach, combining several model variations to construct a more robust final model. However, their paper did not resolve the specific details of this process.

An Ensemble strategy based on Stacked Generalization was adopted by Morio et al. [MMOM20]. This method used multiple classifier predictions as inputs for a Meta Estimator. Hyperparameter Search and Cross-Validation were performed to optimize the Meta Estimator, with the Learning Rate and Dropout Ratio being the key Hyperparameters. The final number of models generated was determined by multiplying the number of pretrained Transformer models by the number of Hyperparameter sets and K-folds. Only those with the best validation scores were selected among these models for further processing. These selected models then predicted their validation folds on the training data.

The predicted validation folds were concatenated for each Hyperparameter set, creating meta-features to train the Meta Estimator. The labels were predicted during testing using fine-tuned models with the best Hyperparameters. The trained Meta Estimator used these predicted labels to yield the final prediction.

Lastly, Raj et al. [RJR$^+$20] implemented an Ensemble strategy at various stages of their model. The level-2 and level-1 classifiers were used in Ensembles to generate the final prediction.

3.3.3 Analysis of the Rest

After looking deeply at the best five submissions, all other submissions are analyzed in this section.

Preprocessing

Grigorev and Ivanov [GI20] implemented Undersampling for the over-represented classes *Loaded Language* and *Name Calling, Labeling*, setting Undersampling ratios of 0.2 and 0.5, respectively.

In their Preprocessing phase, Blaschke et al. [BKT20] extracted Named Entity Tags, including predictions for nationalities, religious or political groups, and geographic entities. They introduced question features and rhetorical question features.

Bairaktaris et al. [BSA20] also used Preprocessing techniques to create four lists containing countries, politically related words, religions, and slogans. During Preprocessing, propaganda spans were scanned for list items replaced with corresponding tags. Named Entity Recognition was applied, replacing politicians' names with a *PERSON* tag providing the best results.

Kaas et al. [KTP20] incorporated 54 hand-crafted features into their model, identifying five as particularly performance-driving. These included counting the reappearance of stemmed one-word spans and spans longer than one word, counting the number of words in a span, and calculating the inverse uniqueness of words in a span.

Krishnamurthy et al. [KGY20] extended BERT with Emotional Intensity Analysis of propaganda spans and extracted 73 word-level psycholinguistic features from the Linguistic Inquiry and Word Count (LIWC) lexicon [PBJB15].

Kim and Bethard [KB20] experimented with the span context by modeling inputs with only the propaganda span, the parent sentence, and the sentences before and after the propaganda span.

Next, Altiti et al. [AAO20] preprocessed the spans by removing punctuation and special symbols, performing Tokenization, and cleaning contractions.

Martinkovic et al. [MPS20] reduced the number of input tokens by removing Tweet footers, timestamps, web surveys, hyperlinks, advertisements, emoji characters, Twitter[3] mentions, and substituting unicode quotation marks, apostrophes, and hyphens with their ASCII equivalents.

Jiang et al. [JGM20] lowercased and tokenized the propaganda spans for their BERT-based model. To tackle the imbalanced dataset, they undersampled Majority Classes and oversampled Minority Classes to 400 examples per class.

In their Preprocessing phase, Patil et al. [PSA20] , removed non-ASCII characters, performed UTF-8 conversion, lowercasing, stemming, and removed trailing white spaces, newlines, and stopwords. Their feature extraction process involved extracting contextual, content-, and context-based metadata.

Different text-splitting approaches were tested by Paraschiv et al. [PCD20]. However, no approach was able to better the results. Therefore only the exact propaganda span was used as input for their BERT-based model. The authors used Masked Language Modeling on two corpora to further train the BERT model on propagandistic spans.

Paraschiv et al. [PCD20] used only the exact propaganda span as input for their BERT-based model, which was further trained on two corpora containing 8.5 Million Fake News articles[4] and 750000 articles from the Hyperpartisan news corpus [KMS$^+$19].

Li and Xiao [LX20] employed an emotion lexicon to extract word intensity, which was then added to the BERT Embedding. Class weights were introduced to manage class imbalance.

Daval-Frerot and Weis [DFW20] described three approaches to gather more data for the Technique Classification subtask: Backtranslation, Synonym Replacement, and TF-IDF. For their model, they did not use Backtranslation of the lack of a proper translation package. For Transfer Learning, the All-the-news articles[5] dataset was used.

About 3000 additional training samples were created by Kranzlein et al. [KBG20] by randomly replacing verbs, nouns, and adjectives with synonyms. Part-of-Speech Tags, Named Entity Tags, and Keyword Frequency were used as hand-crafted features.

Arsenos and Siolas [AS20] performed minor text alterations by removing most punctuation marks.

Ermurachi and Gifu [EG20] eliminated stopwords and special characters, cleaned initial and ending white spaces, and lowercased the text. They also used Bag-of-Words and TF-IDF during Feature Engineering.

Petee and Palmer [PP20] introduced a primary classifier for a specific task using two features: Bag-of-Words and the average of all the words in the section. Then more

[3] https://twitter.com (last accessed: 03 June 2023)

[4] https://github.com/several27/FakeNewsCorpus (last accessed: 04 January 2023)

[5] https://components.one/datasets/all-the-news-articles-dataset (last accessed: 05 January 2023)

features were added, including information about Named Entities and their sentiment across three dimensions: valence, arousal, and dominance.

Finally, Verma et al. [VMC20] used character-level annotations to create word-level inputs, kept track of the character-level indices of each word, removed empty sentences, and combine sentences that were part of the same propaganda span. For their BERT and ELMo-based models, they formatted the data as a sequence of input and output and trained their model by One-hot Encoding the tokenized words. They generated sentence representations using a pretrained Word Embedding with LSTM and BERT.

System Setup

The Cost-Sensitive Learning approach, utilized by Grigorev and Ivanov [GI20], addressed class imbalance by assigning inverse weights relative to the specific class proportion in the dataset. This approach resulted in non-zero F1 scores across all classes.

The system proposed by Blaschke et al. [BKT20] primarily focused on the *Repetition* class. It was achieved by breaking down the prediction generation into subsystems, including *Base* and *Repetition Models*. A Multilayer Perceptron provided the final prediction. If both *Base* and *Repetition Models* failed to predict the *Repetition* class, a third model would reclassify the fragment into one of the remaining 13 classes.

Implementing BERT outperformed the traditional Machine Learning proposed by Bairaktaris et al. [BSA20]. The most effective approach involved labeling the data with *NATION*, *RELIGION*, *POLITICS*, or *SLOGANS*, which yielded improved results, especially in some Minority Classes.

The complete model proposed by Kaas et al. [KTP20] consists of three building blocks. The first building block is a BERT model, where input was only the actual propaganda span without context. A 10-fold Stratified Learning Strategy was used to obtain the best model. Ten stratified splits were created from training data. The fine-tuned model was fed with the folds until no further loss increase was encountered on the previously created test split from the training data. A Linear Layer was modeled on top to get a 14-dimensional output vector representing the 14 classes. The second model was a Logistic Regression model used with the extracted features. The result of this model was then concatenated with the output of the BERT model and the extracted features. The last component, a Fully-connected Feedforward Network with three Hidden Layers, took the output from BERT, the hand-crafted feature, and the output of the Logistic Regression as its input and returned the final prediction.

In the approach by Krishnamurthy et al. [KGY20], various feature sets were concatenated and processed through a Dense Layer. The output layer of the system was a Fully-connected Layer with Softmax Activation. Notably, including the LIWC Lexicon [PBJB15] did not improve the results, while adding information about emotions did improve the final result.

Kim and Bethard [KB20] exploited different variations of BERT and RoBERTa in their submission. The final model combined BERT-large and RoBERTa-base without a context feature.

Altiti et al. [AAO20] selected BERT as their optimal model after testing various other options, including a simple Neural Network and a Convolutional Neural Network.

The custom model submitted by Martinkovic et al. [MPS20] utilized ELMo word representations, a single Bi-LSTM Encoding Layer, a Self-Attention Layer, and a Linear Layer for decoding.

Jiang et al. [JGM20] applied Bagging on nine BERT models, each trained on different subsets of the training data. The most frequently occurring prediction was selected.

Patil et al. [PSA20] built a model that combined input spans processed through BERT and Logistic Regression models. The output from these models was then fed into another Logistic Regression model to generate the final prediction.

To pretrain BERT, Paraschiv et al. [PCD20] used two additional corpora to sensibilize BERT for linguistic structures of propagandistic spans. The corpora were used to train BERT for 2 million steps. The resulting pretrained BERT model is extended with a Dense Layer funnel of 768, 256, and 14, followed by a final Softmax Layer. If a span overlapped with another span, the two overlapping spans were not labeled with the same class. The authors assigned distinct classes in such cases.

Li and Xiao [LX20] implemented a three-part approach consisting of a BERT model with Cost-Sensitive Learning, a model for emotional features, and a Logistic Regression model with various continuous and boolean features.

Daval-Frerot and Weis [DFW20] fine-tuned a simple BERT model and compared it to a similar bidirectional LSTM network to avoid Overfitting. Their model was based on Unsupervised Data Augmentation [XDH$^+$20].

Arsenos and Siolas [AS20] trained a Multilayer Perceptron using pretrained word vectors from the Word2Vec model [MCCD13] and Word Embeddings from GloVe [PSM14]. Before being used, the data was processed through a bidirectional LSTM.

Ermurachi and Gifu [EG20] used traditional Machine Learning, specifically Bag-of-Words, and TF-IDF, to create a feature set for a Random Forest [Ho95] classifier.

Lastly, the Multi-System-Framework by Verma et al. [VMC20] utilized BERT with a weighted loss. The output was fed into three systems: a Linear Layer with Dropout, a Linear Layer with Multi-Sample Dropout, and a Convolutional Neural Network with Multi-Sample Dropout. The outputs of all three were aggregated to form the final prediction.

Postprocessing

As described in the System Setup, Blaschke et al. [BKT20] leveraged Postprocessing to overwrite the predictions from their model, if applicable. Also, the authors observed the input fragments for duplicates, and if found, a model was used to label the duplicate instance [BKT20].

After Ensembling their different learning models, Li and Xiao [LX20] used Rule-based Correction and Reinforcement to reassign classes if the *Repetition* class is predicted [LX20].

Kranzlein et al. [KBG20] used a LSTM model, in which BERT Embeddings and all features were fed in.

Ensemble

Kaas et al. [KTP20] ensemble the results of the BERT model, the extracted features, and the result of the Logistic Regression to obtain a better-performing model from two weak learners.

The Ensemble Model proposed by Kim and Bethard [KB20] is based on an Average Ensemble. The combined models were a BERT-large and a RoBERTa-base model.

A completely different approach in Ensemble Learning is used by Li and Xiao [LX20]. Instead of averaging their different models' predictions, they took the *Repetition* class from the Logistic Regression Model, the Majority Classes from the fine-tuned BERT model, and the Minority Classes from the Cost-Sensitive Learning approach.

Daval-Frerot and Weis [DFW20] repeated their training multiple times with minor variations. The results were aggregated into an Ensemble not further described in their paper.

After the BERT model, Verma et al. [VMC20] ensemble their prediction from their Linear Layer with Dropout, a Linear Layer with Multi-Sample Dropout, and a Convolutional Neural Network with Multi-Sample Dropout.

CHAPTER 4

CRISP-DM

4.1 Business Understanding

The European research on disinformation and propaganda [BHL+21] is used to elaborate a comprehensive understanding of the Business Context and its potential stakeholders involved.

4.1.1 Business Context 1: Addressing Media Distrust

Examining the role of propaganda and disinformation in fostering distrust in media information is part of the first identified Business Context. By identifying the key propaganda techniques used and making the propaganda usage visible, this thesis aims to improve media credibility, transparency, and trustworthiness. Those improvements enable the public to better distinguish between reliable sources and those that promote disinformation, fostering a more informed and engaged society [BHL+21]. Possible stakeholders could be news organizations, journalists, media regulatory bodies, social media platforms, fact-checking organizations, educational institutions, and the general public.

4.1.2 Business Context 2: Preserving Democratic Processes

Identify and analyze disinformation and propaganda content to protect democratic processes in the European Union and its member states. By understanding the techniques and strategies employed, policymakers and stakeholders can develop targeted countermeasures that minimize disinformation's negative impact on democratic processes. Possible stakeholders could be: European Union institutions, national governments, policymakers, political parties, electoral commissions, media organizations, civil society organizations, and the general public.

4.1.3 Business Context 3: Restoring Trust in Institutions

By analyzing the media coverage of the attack on Ukraine, propaganda techniques that erodes trust in institutions could be revealed. Finding propaganda in state-published media could raise awareness towards manipulation-free communication. Doing so aims to help restore citizens' confidence in these institutions, fostering a more stable and functional democratic system. Possible stakeholders are National governments, European Union institutions, policymakers, media organizations, journalists, social media platforms, civil society organizations, and the general public.

4.1.4 Business Context 4: Supporting Media Literacy

By providing a tool to identify propaganda in online news articles, this thesis contributes to media literacy initiatives. By helping users better understand and critically evaluate the information they consume, an informed and discerning public, better equipped to navigate the complex media landscape, is fostered. The possible stakeholders are media literacy organizations, educational institutions, teachers, students, researchers, media watchdog groups, social media platforms, non-governmental organizations promoting media literacy, and citizens interested in becoming more informed and the general public.

4.1.5 Conclusion on Business Understanding

In total more than four Business Contexts were identified during the analysis of the European research on disinformation and propaganda [BHL$^+$21]. However, not all were suitable to be thematized within the scope of the thesis.

4.2 Data Understanding

This chapter outlines the steps necessary to entirely understand the data used for the project, including Data Collection, Exploration, and Quality Assessment.

The data used can be split into two distinct datasets. The first dataset trains, optimizes and evaluates the Deep Learning model. After model training, the second dataset is used to analyze the news articles published before and during the Ukraine crisis.

4.2.1 Propaganda Techniques Corpus

To be able to train the Propaganda Detection Model, an annotated dataset for Supervised Learning is used.

Data Collection

The Propaganda Techniques Corpus is a dataset of texts annotated with 18 fine-grained propaganda techniques. Six professional annotators manually annotated the corpus, and specific underrepresented techniques were merged for simplification. The dataset is organized into training, development, and test sets, with articles in plain-text format. The dataset contains 446 articles from 48 news outlets. The corpus enables three tasks: Propaganda Span Identification, Propaganda Technique Labeling, and Fragment Level Classification. For each task, the propaganda labels are provided [DSMYBC+19].

Data Exploration

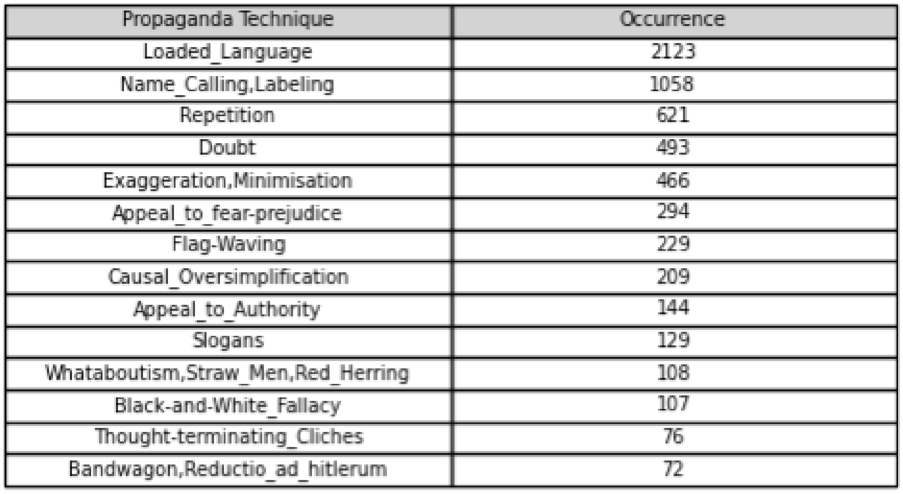
The Propaganda Technique Corpus is divided into training, development, and test sets. However, the test set is not annotated, as it was created to organize public competitions. Consequently, the test set was evaluated by submitting model inference to a website that calculated the final evaluation performance. This option was turned off after the challenge ended. For this thesis, this implies that it is not feasible to evaluate the results of the test set. The challenge summary published the Evaluation Metrics for the development set. As a result, the training set will be utilized to train the model. In contrast, the development set will be employed for evaluation and model ranking based on the challenge participants' results on the development dataset.

The training dataset comprises 6128 distinct propaganda spans, with two classes, *Loaded Language* and *Name Calling/Labeling* - being significantly overrepresented, having 2123 and 1058 instances, respectively. Additionally, *Repetition* (621 instances), *Doubt* (493 instances), and *Exaggeration/Minimization* (466 instances) are also prevalent, forming the five Majority Classes in the dataset. Conversely, there are nine Minority Classes, each accounting for less than 5% coverage. These include *Appeal to fear-prejudice* (294 instances), *Flag-Waving* (229 instances), *Causal Oversimplification* (209 instances), *Appeal to Authority* (144 instances), *Slogans* (129 instances), *Whataboutism/Straw Men/Red Herring* (108 instances), *Black-and-White Fallacy* (107 instances), *Thought-terminating Cliches* (76 instances), and *Bandwagon/Reductio ad Hitlerum* (72 instances). The findings are summarized in Figure 4.1. Next, Figure 4.2 shows the average span length of the different propaganda techniques. The shortest spans are within *Loaded Language*, *Name Calling/Labeling*, *Repetition*, *Slogans*, and *Thought-terminating Cliches*, while the longest are within the *Doubt*, *Appeal to Authority*, *Causal Oversimplification*, and *Black-and-White Fallacy* techniques.

The development dataset contains 1063 examples of propaganda with a nearly identical distribution of the occurrences of the propaganda techniques.

Data Quality Assessment

The Propaganda Technique Corpus is an imbalanced Multi-label Multi-class Dataset, which raises challenges in developing an accurate Propaganda Detection Model. However, the overall data quality is adequate, as Da San Martino et al. [DSMYBC+19] have already implemented measures to maintain high quality. Despite these efforts, the articles have not been cleaned, so the text may still contain non-relevant information, such as author names or Twitter cards.



Propaganda Technique	Occurrence
Loaded_Language	2123
Name_Calling,Labeling	1058
Repetition	621
Doubt	493
Exaggeration,Minimisation	466
Appeal_to_fear-prejudice	294
Flag-Waving	229
Causal_Oversimplification	209
Appeal_to_Authority	144
Slogans	129
Whataboutism,Straw_Men,Red_Herring	108
Black-and-White_Fallacy	107
Thought-terminating_Cliches	76
Bandwagon,Reductio_ad_hitlerum	72

Figure 4.1: Propaganda Technique Distribution for Training

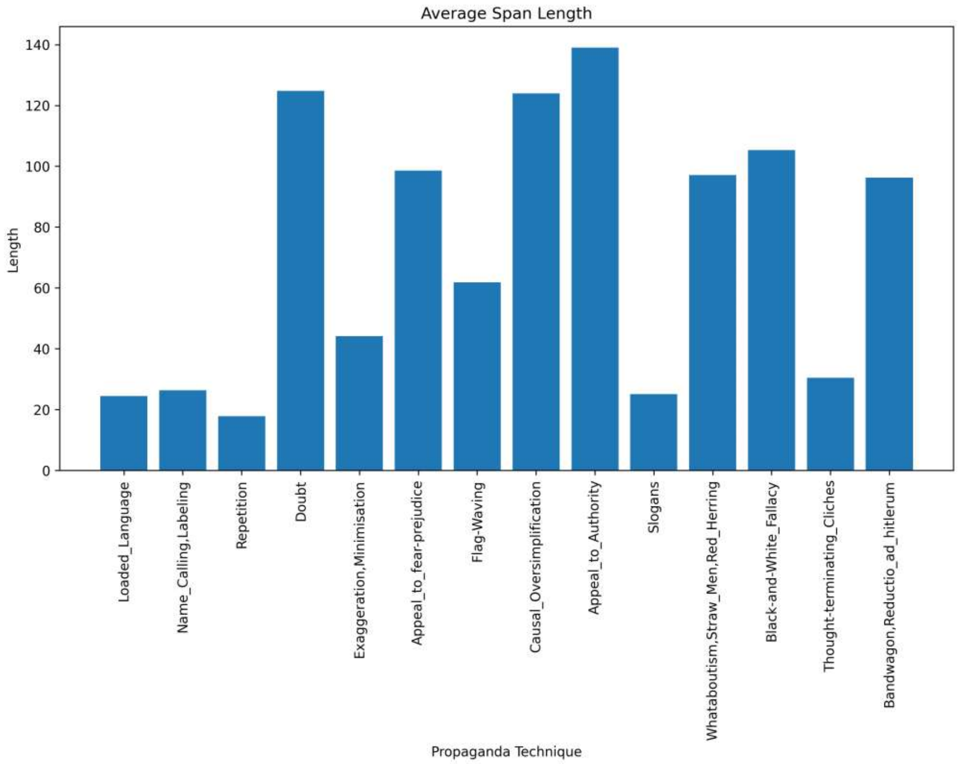


Figure 4.2: Average Span Length of Propaganda Spans for Training

4.2.2 Analysis Dataset

The Analysis Dataset is used to analyze the self-gathered articles from Russian and American sources.

Data Collection

The Analysis Dataset comprises these specific articles after developing the model to analyze propaganda usage in American and Russian news articles. The news articles were gathered from ten different sources, with five for each country. The American websites scraped include ABC News[1], CBS News[2], CNN International[3], Fox News[4], and Politico[5].

The Russian websites scraped are News Front[6], Novye Izvestia[7], RT[8], Sputnik News[9], and Tass Russian News Agency[10]. Each article's title, subtitle, body, and creation date were saved. The Data Collection ranges from 01 January 2022 to 26 May 2023. Initially, 14689 American news articles and 55231 Russian news articles were scraped.

Data Exploration

Further information on the propaganda spans will be provided in Chapter 5 while evaluating the results.

Data Quality Assessment

For Politico and News Front, the scraped articles range from 23 February 2022 to 24 April 2023. The scraping of articles before and after this range failed multiple times. Since the difference in articles was only around 200, respectively, the decision was made to accept this minor cut in Data Quality.

4.3 Data Preparation

The Data Preparation stage of the CRISP-DM methodology is essential for transforming raw data into a format suitable for analysis and modeling. This chapter outlines the critical steps in preparing the project's data, including Data Cleaning, Integration, Transformation, and Feature Engineering.

[1] https://abcnews.go.com (last accessed: 08 July 2022)
[2] https://www.cbsnews.com (last accessed: 08 July 2022)
[3] https://edition.cnn.com (last accessed: 08 July 2022)
[4] https://www.foxnews.com (last accessed: 08 July 2022)
[5] https://www.politico.com (last accessed: 08 July 2022)
[6] https://en.news-front.info (last accessed: 08 July 2022)
[7] https://en.newizv.ru (last accessed: 08 July 2022)
[8] https://www.rt.com (last accessed: 08 July 2022)
[9] https://sputniknews.com (last accessed: 08 July 2022)
[10] https://tass.com (last accessed: 08 July 2022)

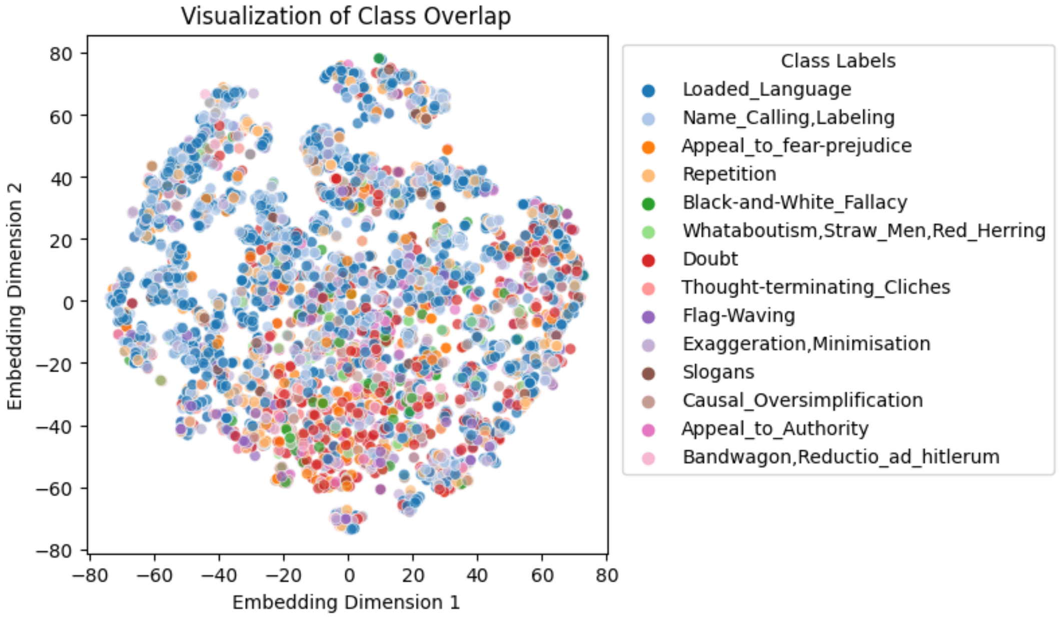


Figure 4.3: Visualization of the Propaganda Techniques' Overlap

4.3.1 Propaganda Technique Corpus

The first step to preparing the date for model training is to address any identified quality issues during the Data Understanding stage.

Data Cleaning

The Propaganda Technique Corpus appears free of issues concerning missing values, duplicate articles, or data entry errors.

Da San Martino et al. [DSMYBC+19] suggest that eliminating negative samples in their experiments improves outcomes. Outlier Detection for specific propaganda classes could help remove propaganda spans that generate excessive noise. Visualizing the overlap of the 14 propaganda techniques reveals that the classes' Embeddings significantly overlap, making it hard to identify all 14 clusters. The extensive overlap of propaganda techniques can be observed in Figure 4.3. Because professional annotators labeled the Propaganda Technique Corpus, it is reasonable to assume that the various propaganda classes are annotated accurately. Attempting to remove overlapping spans results in losing approximately 5,000 training examples. While the data may appear cleaner, Minority Classes are lost, leaving only the Majority Classes left. Consequently, overlapping spans are not removed.

Data Integration

Since the Propaganda Technique Corpus is the only source used for training, there is no need to align the schema of different sources. Also, Data Concatenation and Entity Resolution are not relevant here.

Data Transformation

Transforming the data is necessary to ensure it is suitable for analysis and modeling. In the specific case of using Language-based Transformer Models like RoBERTa, the spans, which are categorical variables, must be encoded to be usable for the model. Normalization, which is the process of scaling numeric variables, is unnecessary since no numeric variables exist in the dataset. For the same reason, Date and Time Conversion is not needed.

Feature Engineering

Feature Engineering involves the creation of new variables or features that may improve the performance of the Data Mining models. In the case of the Propaganda Technique Corpus, a new context feature was introduced, which takes in the tokens left and right of the propaganda span until the whole sentence carrying the propaganda span is found.

The second crafted feature is the length of a propaganda span. As seen in Figure 4.2, the propaganda techniques tend to have different lengths, which could be a helpful indicator for the Propaganda Detection Model.

4.3.2 Analysis Dataset

The Analysis Dataset, a collection of scraped articles from the web, needs more effort to process. The structure and schema of the Propaganda Technique Corpus were used as a guideline to create a suitable dataset to be used for Inference with the Propaganda Detection Model.

Data Cleaning

After scraping the Analysis Dataset, the content is split into the title, an optional subtitle, and the article content. Further, a publishing date, a unique identifier, and an article link were saved during scraping. During scraping, duplicates were ignored, and during Data Cleaning, this was verified. Finally, the Analysis Dataset was checked for non-existing values, but all values were set except for the optional subtitle.

A series of customized functions were introduced to improve text data processing. The method includes the implementation of various text cleaning functions, such as adding whitespaces after periods, removing URLs and emojis, replacing trailing whitespaces, replacing newline and tab characters with spaces, removing non-breaking spaces, and adjusting various types of quotation marks to a uniform style.

The articles are split into sentences, where each line represents one sentence. Furthermore, each sentence is tokenized, ensuring that the tokenized sentences do not exceed a maximum token limit of 256. If a sentence exceeds this limit, it is skipped. Otherwise, the model fails to identify the propaganda spans.

Data Integration

During Data Integration, the articles' titles, subtitles, and contents are concatenated and saved into text files. The unique article id is used as the file's name to identify the articles.

Data Transformation and Feature Engineering

Since just passing whole sentences to identify the propaganda techniques in the Analysis Dataset is impossible, it is necessary to identify the propagandistic spans. This limitation was due to the Propaganda Detection Model being trained on propagandistic spans rather than whole sentences. The Span Identification model developed by Chernyavskiy et al. [CIN20] is used to identify the beginning and ending tokens of the propaganda spans.

The authors treat this task as a Sequence Labeling Problem and use a Begin, Inside, Outside (BIO) tagging format. The RoBERTa model is fine-tuned to predict BIO tags for each token in a sentence. However, since the RoBERTa model does not account for the dependency between predicted labels, a CRF is added as an extra layer. This extra layer helps model the relationship between individual token labels and improves predictions. The RoBERTa-CRF model is trained end-to-end, with the CRF receiving logits for each token and predicting the entire input sequence while considering label dependencies. The CRF works with words, so only tokens that start a word are passed to it, while word continuation tokens are skipped [CIN20].

For Data Transformation and Feature Engineering, the propagandistic spans were identified, BIO tags were generated, and the articles were encoded to fit the numerical input needs of the Transformer models.

4.4 Model

The Propaganda Detection Model was built iterative, and different approaches were tested and evaluated. Whenever a rise in Micro F1-Score is detected, the resulting model is used as the new reference model.

4.4.1 The Baseline Model

In the initial stage of this research, a rudimentary model was developed to establish a foundational benchmark for evaluation. This elementary system possesses the following characteristics:

1. Utilization of propaganda span exclusively as input for the Transformer model.
2. Implementation of RoBERTa as the preliminary Language Model, given its efficacy and superior results during the SemEval Challenge [DSMBCW+20].
3. Adopting the RoBERTa-base model, as the training process for RoBERTa-large is financially expensive during the beginning stages of research.

Training Preparation

Building a Deep Learning system based on Transformer-based Language Models first requires downloading the model and its corresponding Tokenizer and setting up the work environment for Hardware Acceleration, if available.

The training dataset was already prepared during Data Preprocessing, and to evaluate the *Baseline Model*, only the propaganda span was used as model input.

The input is tokenized since Transformer-based Language Models cannot consume characters and strings. The Tokenization process added special tokens and set a maximum allowed length for the input spans. The maximum length parameter of RoBERTa's Tokenizer refers to the maximum number of subwords generated from the input text. Therefore, it is based on the number of subwords, not characters or words. The *Baseline Model* allows for a sequence length of 512 subwords, and the sequence is padded to the maximum length if the input is shorter than the maximum length.

The AdamW optimizer[11] has a Learning Rate of 1e-5 and an Epsilon Rate of 1e-8. Further, a Linear Scheduler without a warm-up is used.

Model Setup

The *Baseline Model* is implemented with PyTorch [PGM+19] and uses a standard training setup. A data loader is used to enumerate through all the batched inputs, and every batch is then passed to the RoBERTa model. The model returns a loss value, calculating an average training loss after all batches are consumed. After every training epoch, a simple evaluation process is executed to evaluate the performance of the current epoch.

Results

The Micro F1-Score is at 0.54 for the *Baseline Model*, but looking at the distinct classes reveals more information. Majority Classes like *Loaded Language*, *Flag-Waving*, *Name Calling/Labeling*, are already predicted relatively well. Even though *Repetition* is within the three most extensive classes, the model struggles to predict the propaganda technique. Also, the model cannot predict four classes at all, namely *Appeal to Authority*, *Bandwagon/Reductio ad hitlerum*, *Black-and-White Fallacy* and *Whataboutism/Straw*

[11] https://pytorch.org/docs/stable/generated/torch.optim.AdamW.html (last accessed: 29 May 2023)

Table 4.1: Classification Report Baseline Model

Class	Precision	Recall	F1-Score	Support
Appeal to Authority	0.0	0.0	0.0	14
Appeal to fear-prejudice	0.32	0.41	0.36	44
Bandwagon/Reductio ad hitlerum	0.0	0.0	0.0	5
Black-and-White Fallacy	0.0	0.0	0.0	22
Causal Oversimplification	0.29	0.28	0.29	18
Doubt	0.42	0.67	0.51	66
Exaggeration/Minimisation	0.52	0.51	0.52	68
Flag-Waving	0.68	0.68	0.68	87
Loaded Language	0.66	0.85	0.75	325
Name Calling/Labeling	0.63	0.76	0.69	183
Repetition	0.38	0.2	0.26	145
Slogans	0.67	0.2	0.31	40
Thought-terminating Cliches	0.25	0.06	0.1	17
Whataboutism/Straw Men/Red Herring	0.0	0.0	0.0	29
Micro F1-Score	0.53	0.58	0.54	1063

Men/Red Herring. All other classes have scores below 0.5. Table 4.1 shows an overview of the results.

This setup is used as a starting point. In the following chapters, only the changes regarding the setup of the *Baseline Model* are described.

4.4.2 Testing Hyperparameter Optimization

This section describes Hyperparameter Tuning for the Learning and Dropout Rate.

Optimizing the Learning Rate

Hyperparameter Tuning is performed on the *Baseline Model* using the Hyperopt library[12]. The first Hyperparameter used for tuning is the Learning Rate. Using a Logarithmic Distribution to find the best Learning Rate for the model, a new Learning Rate is tested with a fresh model, optimizer, and scheduler. The model is trained for a predefined number of epochs. The final training loss and the Micro F1-Score are returned.

Model Changes

The Learning Rate for the *Baseline Model* is changed to 2.4e-05.

[12] http://hyperopt.github.io/hyperopt/ (last accessed: 13 April 2023)

Table 4.2: Classification Report Hyperparameter Model

Class	Precision	Recall	F1-Score	Support
Appeal to Authority	0.09	0.07	0.08	14
Appeal to fear-prejudice	0.26	0.18	0.21	44
Bandwagon/Reductio ad hitlerum	0.0	0.0	0.0	5
Black-and-White Fallacy	1.0	0.09	0.17	22
Causal Oversimplification	0.36	0.22	0.28	18
Doubt	0.42	0.67	0.51	66
Exaggeration/Minimisation	0.47	0.6	0.53	68
Flag-Waving	0.75	0.8	0.78	87
Loaded Language	0.76	0.8	0.78	325
Name Calling/Labeling	0.63	0.84	0.72	183
Repetition	0.4	0.23	0.29	145
Slogans	0.62	0.6	0.61	40
Thought-terminating Cliches	0.38	0.18	0.24	17
Whataboutism/Straw Men/Red Herring	0.14	0.03	0.06	29
Micro F1-Score	0.58	0.58	0.58	1063

Results

The *Hyperparameter Model* (see Table 4.2) has a higher Micro F1-Score (0.58) than the *Baseline Model* (0.54) (see Table 4.1), which indicates better overall performance. For some classes like *Black-and-White Fallacy*, *Causal Oversimplification*, *Exaggeration/ Minimisation*, *Flag-Waving*, *Loaded Language*, *Name Calling*, *Labeling*, *Repetition*, *Slogans*, and *Thought-terminating Cliches*, the *Hyperparameter Model* performs better in terms of F1-Score. Notably, the *Black-and-White Fallacy* class saw a significant increase in Precision from 0.0 to 1.0, though its Recall is still low. The *Baseline Model* has a higher F1-Score for the *Appeal to fear-prejudice* class. Both models have low scores for the *Appeal to Authority* and *Whataboutism/Straw Men/Red Herring* classes, but the *Hyperparameter Model* shows minor improvement. The *Doubt* class has the same F1-Score in both models, with no change in Precision and Recall. *Bandwagon/Reductio ad hitlerum* has all zero values for Precision, Recall, and F1-Score in both models, indicating that neither model could correctly classify any instances of this class.

The Learning Rate has an essential influence on the results but is still insufficient to improve the model dramatically.

4.4.3 Optimizing the Dropout Rate

After identifying the optimal Learning Rate for the *Baseline Model*, the Dropout Rate needs to be modified to prevent Overfitting. By default, the Dropout Rate for the RoBERTa-base model is fixed at 0.1. Again the Hyperopt library is incorporated to search for the optimal Dropout Rate. Such a rate must be able to balance learning time and performance gains feasibly. The model is called *Dropout Model*.

Model Changes

The Dropout Rate for the *Baseline Model* is changed from 0.1 to 0.2.

Results

According to Table 4.3, changing the Dropout Rate to 0.2 increases the Micro F1-Score by 0.02 points to 0.60. The most significant change is that no class is zero-predicted after optimizing the Dropout Rate. In contrast, compared with the *Baseline Model*, the model struggled to predict five Minority Classes. The *Dropout Model* performs better in all metrics for *Appeal to Authority*, *Appeal to fear-prejudice*, *Bandwagon/Reductio ad hitlerum*, and *Causal Oversimplification*. The classes *Black-and-White Fallacy*, *Doubt*, *Name Calling/Labeling*, *Repetition*, *Slogans*, and *Whataboutism/Straw Men/Red Herring* show better performance by the *Dropout Model* in F1-Score, but mixed results in Precision and Recall. Both models' equal or nearly equal performance was shown for *Exaggeration/Minimisation*. In contrast, better performance by the *Dropout Model* in Recall and F1-Score was shown by *Flag-Waving* and *Loaded Language*. Finally, only the *Thought-terminating Cliches* class performs better by the *Dropout Model* in Precision and F1-Score but has lower Recall.

Overall, the *Dropout Model* performs better across most classes regarding F1-Score. However, there are cases, such as *Doubt*, *Name Calling/Labeling*, *Repetition*, *Slogans*, and *Whataboutism/Straw Men/Red Herring*, where the *Hyperparameter Model* offers higher Recall, indicating its relative strength in identifying True Positives, but potentially at the expense of more False Positives. Like the Learning Rate, the Dropout Rate influences the results, and optimizing both Hyperparameters helps improve the *Baseline Model*. Combining both Hyperparameters, the *Dropout Model* is part of the Ensemble Strategy.

4.4.4 Testing Context Addition

Now knowing the optimal Learning and Dropout Rate for the *Baseline Model*, the following experiment performed incorporates a context feature with two different setups tested. The first setup adds the context feature directly during the encoding phase to the propaganda span. The model's input is the concatenated propaganda span with its context, split by special tokens. Since this setup performed drastically worse than the *Baseline Model*, it is discarded.

Model Changes

The second setup defines a custom *Context Model* for Sequence Classification. This model utilizes two RoBERTa models and combines their outputs to perform the classification

Table 4.3: Classification Report Dropout Model

Class	Precision	Recall	F1-Score	Support
Appeal to Authority	0.13	0.21	0.16	14
Appeal to fear-prejudice	0.30	0.39	0.34	44
Bandwagon/Reductio ad hitlerum	1.0	0.6	0.75	5
Black-and-White Fallacy	0.22	0.09	0.13	22
Causal Oversimplification	0.4	0.33	0.36	18
Doubt	0.48	0.58	0.52	66
Exaggeration/Minimisation	0.47	0.59	0.52	68
Flag-Waving	0.71	0.83	0.76	87
Loaded Language	0.73	0.81	0.77	325
Name Calling/Labeling	0.65	0.75	0.70	183
Repetition	0.44	0.25	0.32	145
Slogans	0.69	0.50	0.58	40
Thought-terminating Cliches	0.33	0.06	0.10	17
Whataboutism/Straw Men/Red Herring	0.25	0.03	0.06	29
Micro F1-Score			0.60	1063

task. One model takes the propaganda span as input, while the other consumes the context.

The *Context Model* processes the context and span inputs through each RoBERTa model. The outputs of these models are then passed through a Linear Layer to condense the information. Subsequently, the outputs of both models are concatenated and passed through a Dropout Layer for Regularization. A Linear Classifier then processes the concatenated output.

Results

As shown in Table 4.4, the overall Micro F1-Score did not increase. Interestingly, the classes *Appeal to Authority*, *Appeal to fear-prejudice*, *Name Calling/Labeling*, *Loaded Language*, and Flag-Waving. *Thought-terminating Cliches* and *Whataboutism/Straw Men/Red Herring* have unchanged performance. *Bandwagon/Reductio ad hitlerum* is the class that benefits the most, with an F1-Score increase from 0.57 to 0.75.

Unfortunately, the *Context Model* is resource-intensive, making adding further features and functions expensive. Due to running simultaneously two instances of RoBERTa models at the same time, the computational limits of the hardware are reached quickly. Trying to use a larger RoBERTa model or adding more features results in crashes due to out-of-memory exceptions. Considering these limitations, the performance does not increase significantly to justify further optimizing the *Context Model*. Therefore, the optimized *Baseline Model* is used for further experiments. Variations of the *Context Model* are part of the Ensemble Learning strategy.

Table 4.4: Classification Report Context Model

Class	Precision	Recall	F1-Score	Support
Appeal to Authority	0.50	0.21	0.30	14
Appeal to fear-prejudice	0.37	0.43	0.40	44
Bandwagon/Reductio ad hitlerum	1.00	0.40	0.57	5
Black-and-White Fallacy	0.50	0.05	0.08	22
Causal Oversimplification	0.23	0.44	0.30	18
Doubt	0.52	0.48	0.50	66
Exaggeration/Minimisation	0.44	0.54	0.49	68
Flag-Waving	0.68	0.90	0.77	87
Loaded Language	0.68	0.82	0.75	325
Name Calling/Labeling	0.68	0.74	0.71	183
Repetition	0.40	0.23	0.29	145
Slogans	0.65	0.42	0.52	40
Thought-terminating Cliches	0.25	0.06	0.10	17
Whataboutism/Straw Men/Red Herring	0.50	0.03	0.06	29
Micro F1-Score			0.60	1063

4.4.5 Adding Context, Span Length, and Averaged Embedding

After dropping the *Context Model* due to high resource costs, the following model is built on top of the *Baseline Model*. During training, the Classification Token is extracted from the last Hidden Layer, the Averaged Embedding for non-padding tokens is calculated from the remaining Intermediate Layers, and the length of the span is calculated. The features are then concatenated and handed through a Linear Classifier. This model is called *Averaged Embedding Model*.

Model Changes

The most significant change is reintroducing the context feature during Tokenization, the Averaged Embedding for non-padding tokens, using the span length feature, and the concatenation with the Classification Token. Further, Dropout Rates of 0.1 and 0.2 are tested.

Results

The *Averaged Embedding Model* with a Dropout Rate of 0.2 shows a higher Micro F1-Score (0.62) compared to the model with a Dropout Rate of 0.1 (0.61), indicating slightly better overall performance (see Table 4.5 and Table 4.6). This model, with a Dropout of 0.2, shows improved F1-Scores for *Appeal to fear-prejudice*, *Bandwagon/Reductio ad hitlerum*, *Exaggeration/Minimisation*, *Flag-Waving*, *Loaded Language*, *Name Calling/Labeling*, *Repetition*, and *Slogans* classes compared to the *Averaged Embedding Model* with Dropout of 0.1.

Table 4.5: Classification Report Averaged Embedding Model with Dropout of 0.1

Class	Precision	Recall	F1-Score	Support
Appeal to Authority	0.30	0.50	0.38	14
Appeal to fear-prejudice	0.28	0.48	0.36	44
Bandwagon/Reductio ad hitlerum	0.24	0.80	0.36	5
Black-and-White Fallacy	0.25	0.09	0.13	22
Causal Oversimplification	0.41	0.39	0.40	18
Doubt	0.51	0.68	0.58	66
Exaggeration/Minimisation	0.48	0.59	0.53	68
Flag-Waving	0.76	0.78	0.77	87
Loaded Language	0.78	0.72	0.75	325
Name Calling/Labeling	0.69	0.72	0.70	183
Repetition	0.44	0.35	0.39	145
Slogans	0.59	0.55	0.57	40
Thought-terminating Cliches	0.88	0.41	0.56	17
Whataboutism/Straw Men/Red Herring	0.40	0.21	0.27	29
Micro F1-Score			0.61	1063

On the other hand, *Appeal to Authority*, *Black-and-White Fallacy*, *Causal Oversimplification*, *Thought-terminating Cliches*, and *Whataboutism/Straw Men/Red Herring* classes perform worse in the model with a Dropout of 0.2. Especially noteworthy is the *Whataboutism/Straw Men/Red Herring* class, which has all zero values for Precision, Recall, and F1-Score in the Dropout of 0.2 model, indicating that this model could not correctly classify any instances of this class. The *Doubt* class has the same F1-Score in both models, despite changes in Precision and Recall.

While a Dropout Rate of 0.2 worked better for the *Baseline Model*, the *Averaged Embedding Model* generalizes better with a Dropout Rate of 0.1.

Comparing the Hyperparameter optimized *Baseline Model* (*Dropout Model* see Table 4.3) with the better performing *Averaged Embedding Model* with Dropout of 0.1, the following conclusions can be made: With the classes *Appeal to Authority*, *Appeal to fear-prejudice*, *Black-and-White Fallacy*, *Causal Oversimplification*, *Doubt*, *Exaggeration/Minimisation*, *Flag-Waving*, *Repetition*, *Thought-terminating Cliches*, and *Whataboutism/Straw Men/Red Herring* the performance is better with the *Averaged Embedding Model*. In contrast, the remaining classes are better with the *Dropout Model*.

Variations of the *Averaged Embedding Model* are part of the later described Ensembling Strategy.

Table 4.6: Classification Report Averaged Embedding Model with Dropout of 0.2

Class	Precision	Recall	F1-Score	Support
Appeal to Authority	0.23	0.21	0.22	14
Appeal to fear-prejudice	0.32	0.52	0.40	44
Bandwagon/Reductio ad hitlerum	0.50	0.60	0.55	5
Black-and-White Fallacy	0.50	0.05	0.08	22
Causal Oversimplification	0.25	0.33	0.29	18
Doubt	0.55	0.61	0.58	66
Exaggeration/Minimisation	0.47	0.62	0.54	68
Flag-Waving	0.76	0.83	0.79	87
Loaded Language	0.75	0.78	0.76	325
Name Calling/Labeling	0.67	0.75	0.70	183
Repetition	0.50	0.32	0.39	145
Slogans	0.68	0.68	0.68	40
Thought-terminating Cliches	0.75	0.18	0.29	17
Whataboutism/Straw Men/Red Herring	0.00	0.00	0.00	29
Micro F1-Score			0.62	1063

4.4.6 Testing Cost-Sensitive Learning

Even though the model's abilities to classify the Minority Classes increased considerably compared to the *Baseline Model*, the model still struggles to classify at least seven propaganda techniques with a Micro F1-Score greater than 0.5. Cost-Sensitive Learning is introduced to the *Average Embedding Model* to tackle this problem.

Model Changes

For each propaganda technique, a class weight is calculated by counting the occurrences of the distinct techniques in the training dataset. Then the inverse of class frequencies is computed and normalized so that the class weights sum up to 1. Finally, the Cross-Entropy Loss Function is modified to consume the class weights.

Results

Applying Cost-Sensitive Learning has no positive impact on classifying Minority Classes. The model's overall Micro F1-Score is worse, while simultaneously, the model is overfitting quicker.

4.4.7 Testing Undersampling/Oversampling

Since introducing Cost-Sensitive Learning failed to improve the model's ability to predict the Minority Classes, another attempt was made by testing Undersampling the Majority Classes and Oversampling the Minority Classes. This way, equal distribution of the 14

Table 4.7: Classification Report Sampling Model

Class	Precision	Recall	F1-Score	Support
Appeal to Authority	0.30	0.14	0.19	65
Appeal to fear-prejudice	0.32	0.51	0.40	65
Bandwagon/Reductio ad hitlerum	0.67	0.75	0.70	75
Black-and-White Fallacy	0.46	0.22	0.30	72
Causal Oversimplification	0.52	0.27	0.36	81
Doubt	0.44	0.45	0.44	83
Exaggeration/Minimisation	0.37	0.44	0.40	68
Flag-Waving	0.53	0.76	0.62	78
Loaded Language	0.63	0.56	0.59	88
Name Calling/Labeling	0.58	0.71	0.64	78
Repetition	0.48	0.40	0.44	80
Slogans	0.58	0.51	0.54	82
Thought-terminating Cliches	0.27	0.27	0.27	78
Whataboutism/Straw Men/Red Herring	0.37	0.54	0.44	70
Micro F1-Score			0.47	1063

different propaganda classes is achieved. While there are some caveats, like falsifying the real-world situation by skewing the real-world distribution of propaganda technique usage, the approach could improve the Micro F1-Score by giving the Minority Classes a chance to be better identified.

Model Changes

An Imbalanced Dataset Sampler is introduced to the *Average Embedding Model* and the *Dropout Model*, which calculates each propaganda technique's ideal Over- and Undersampling factor. This way, the training data is manipulated to have about 437 training samples per class.

Results

Introducing Undersampling and Oversampling to the Propaganda Technique Corpus yields no further improvements regarding the total Micro F1-Score. While the classes *Appeal to fear-prejudice*, *Bandwagon/Reductio ad hitlerum*, *Black-and-White Fallacy*, and *Whataboutism/Straw Men/Red Herring* perform better with the *Sampling Model*, all other classes are predicted better with the *Averaged Embedding Model* (see Table 4.7 and Table 4.5). The approach is discarded and not used for Ensembling.

4.4.8 Testing Data Augmentation

As previous experiments showed, manipulating the existing data only works up to a specific level. Bringing in new features like the length of the spans and the span's context was previously the most effective way to squeeze out at least some gains in performance.

The next concept tested with the Propaganda Technique Corpus is introducing Data Augmentation as proposed by Daval-Frerot and Weis [DFW20]. During their research, they could not test Backtranslation since back in 2020, and there was no reasonable way to generate those, as they state.

Model Changes

No direct model changes were made. To generate more examples of the Minority Classes, the examples of those were translated into German. In the next step, the examples were translated back into English. This way, examples with minor semantic changes are generated.

Results

While the generation of new training examples led to the best model proposed by Jurkiewicz et al. [JBKG20], generating new training examples with the help of Backtranslation did not help to increase the overall predictive power of the Propaganda Detection Models (see Table 4.8). While the *Augmentation Model* can predict the classes *Appeal to fear-prejudice*, *Bandwagon/Reductio ad hitlerum*, *Black-and-White Fallacy*, *Causal Oversimplification*, and *Loaded Language* better than the *Averaged Embedding Model*, there is no noteworthy increase in performance. Still, due to the variation of input data, an *Augmented Model* is used for Ensembling.

4.4.9 Testing Part-of-Speech Tags

Mapping Part-of-Speech Tags such as nouns, verbs, and adjectives to a continuous vector space can help a model understand the grammatical structure and relationships between words in a sentence, thus improving its performance in predicting propaganda spans.

Model Changes

During the Preprocessing phase, Part-of-Speech Tags are extracted and passed to the model. An additional layer consumes the tags. Then the Part-of-Speech Tags are combined with the Embeddings from the *Dropout Model* and *Averaged Embedding Model*.

Results

Again no performance gains are made (see Table 4.9). Furthermore, most classes decreased performance, and *Thought-terminating Cliches* and *Whataboutism/Straw Men/Red Herring* are not predicted. Only *Bandwagon/Reductio ad hitlerum* benefited from Part-

Table 4.8: Classification Report Augmentation Model

Class	Precision	Recall	F1-Score	Support
Appeal to Authority	0.09	0.14	0.11	14
Appeal to fear-prejudice	0.32	0.52	0.39	44
Bandwagon/Reductio ad hitlerum	0.33	0.80	0.47	5
Black-and-White Fallacy	0.21	0.14	0.17	22
Causal Oversimplification	0.53	0.44	0.48	18
Doubt	0.60	0.47	0.53	66
Exaggeration/Minimisation	0.49	0.56	0.52	68
Flag-Waving	0.78	0.67	0.72	87
Loaded Language	0.72	0.82	0.77	325
Name Calling/Labeling	0.62	0.81	0.70	183
Repetition	0.36	0.14	0.20	145
Slogans	0.54	0.50	0.52	40
Thought-terminating Cliches	0.29	0.12	0.17	17
Whataboutism/Straw Men/Red Herring	0.29	0.17	0.22	29
Micro F1-Score			0.59	1063

of-Speech Tags, making the model irrelevant for further evaluation (see Table 4.5).

4.4.10 Training Additional Models for Ensembling

In the previous steps, different experiments were conducted to find techniques that would be beneficial in improving the Micro F1-Score of the Propaganda Detection Model. Since most failed by making the model worse or did not add significant performance gains, the next possible step is to create multiple weak models and combine them into a single robust predictor. The previously created model setups are trained with different models like BERT, OPT, and RoBERTa with different Hyperparameters and model sizes. To name one example: A combination of the RoBERTa-base model with 125 million parameters and a Hidden Size of 769 could yield performance gains in combination with RoBERTa-large, which has 355 million parameters and a Hidden Size of 1024 [ZWYJ21].

Since generating and combining an indefinite number of models is not feasible, a model qualified for Ensemble Learning must exhibit the following characteristics:

1. Performance of at least 0.57 Micro F1-Score
2. No missed class during evaluation on the validation set

As Table 4.10 shows, nine models with different setups are used for Ensembling Learning. Six models are based on RoBERTa, two make use of OPT, and the last is BERT based.

Table 4.9: Classification Report Part-of-Speech Tags

Class	Precision	Recall	F1-Score	Support
Appeal to Authority	0.06	0.07	0.07	14
Appeal to fear-prejudice	0.32	0.41	0.36	44
Bandwagon/Reductio ad hitlerum	1.00	0.40	0.57	5
Black-and-White Fallacy	0.25	0.05	0.08	22
Causal Oversimplification	0.37	0.39	0.38	18
Doubt	0.42	0.52	0.47	66
Exaggeration/Minimisation	0.39	0.60	0.48	68
Flag-Waving	0.78	0.77	0.77	87
Loaded Language	0.73	0.76	0.75	325
Name Calling/Labeling	0.68	0.73	0.70	183
Repetition	0.40	0.28	0.33	145
Slogans	0.51	0.53	0.52	40
Thought-terminating Cliches	0.00	0.00	0.00	17
Whataboutism/Straw Men/Red Herring	0.00	0.00	0.00	29
Micro F1-Score			0.58	1063

Except for the *Averaged Embedding Models* (Dropout=0.1), the other models' Dropout Rate is fixed at 0.2. For seven models, the Learning Rate is set at 2.4e-5, except for the two *Context Models*, where the same model architecture is used first with a Learning Rate of 3e-5 and second with 2e-5.

4.4.11 Ensembling the Models

For Ensembling Learning, two different strategies are tested: Ensemble Averaging and Meta Classification with Model Stacking. Each model is evaluated on the validation dataset. During Inference, the output right before the final Activation Function, called logits, is extracted. This resulting vector contains the probabilities for the 14 propaganda techniques for every analyzed propaganda span.

Testing Ensemble Averaging

During Ensemble Averaging, the average output from multiple prediction models is calculated by extracting the logits from each model. The Softmax Function is then applied to these averaged logits, transforming them into probabilities ranging between 0 and 1. The class with the highest probability is selected from this resulting prediction vector. This selected class represents the model's prediction for the most likely propaganda technique applicable to the given input.

When comparing the Micro F1-Score of the Ensemble Averaging (Micro F1-Score: 0.63) (see Table 4.11 with the highest Micro F1-Score of the nine models, the *RoBERTa-large Dropout Model* (Micro F1-Score: 0.6171) listed in Table 4.10 the difference is small. Seven

Model Type Architecture	Micro F1-Score	Learning Rate	Dropout
RoBERTa-base Context(1)	59.83%	3e-5	0.2
RoBERTa-base Context(2)	59.36%	2e-5	0.2
RoBERTa-base Averaged Embedding	58.14%	2.4e-5	0.1
RoBERTa-large Averaged Embedding	59.92%	2.4e-5	0.1
RoBERTa-large Dropout	61.71%	2.4e-5	0.2
RoBERTa-large Augmented Dropout	57.01%	2.4e-5	0.2
BERT-large Dropout	61.05%	2.4e-5	0.2
OPT-350M Dropout	61%	2.4e-5	0.2
OPT-1.3B Dropout	59%	2.4e-5	0.2

Table 4.10: Trained Models for Ensembling Learning

out of 14 propaganda techniques are classified with less than 0.5 Micro F1-Score, making it a not trustworthy model.

Testing Meta Classification with Model Stacking

In the second approach evaluated during Ensemble Learning, Meta Classification is employed. Initially, logits from the models are stacked and flattened into a two-dimensional array. Subsequently, Ridge Regression is applied using these flattened logits and the corresponding True Labels. Finally, the propaganda techniques of the input spans are predicted using the trained Ridge Regression Classifier model.

Combining the nine weak models with a Ridge Regression Classifier yields drastic improvements in all classes and an overall Micro F1-Score of 0.78. Also, all Minority classes are predicted with more than 0.50 F1-Score, the worst being *Whataboutism/Straw Men/Red Herring* with 0.57 (see Table 4.12). Ten propaganda techniques have F1-Scores above 0.70. Therefore, the Meta Classification with Model Stacking is a way better Propaganda Prediction Model than the Ensemble Averaging Model (see Table 4.11).

Table 4.11: Classification Report of Ensemble Averaging

Class	Precision	Recall	F1-Score	Support
Appeal to Authority	0.11	0.14	0.12	14
Appeal_to_fear-prejudice	0.39	0.45	0.42	44
Bandwagon/Reductio_ad_hitlerum	0.60	0.60	0.60	5
Black-and-White_Fallacy	0.50	0.09	0.15	22
Causal_Oversimplification	0.35	0.50	0.41	18
Doubt	0.57	0.61	0.59	66
Exaggeration,Minimisation	0.51	0.57	0.54	68
Flag-Waving	0.78	0.84	0.81	87
Loaded_Language	0.72	0.85	0.78	325
Name_Calling,Labeling	0.68	0.76	0.72	183
Repetition	0.46	0.25	0.32	145
Slogans	0.70	0.57	0.63	40
Thought-terminating_Cliches	0.36	0.29	0.32	17
Whataboutism/Straw_Men/Red_Herring	0.25	0.07	0.11	29
Micro F1-Score			0.63	1063

Table 4.12: Classification Report Meta Classification Model

Class	Precision	Recall	F1-Score	Support
Appeal to Authority	1.00	0.71	0.83	14
Appeal to fear-prejudice	0.73	0.55	0.62	44
Bandwagon/Reductio ad hitlerum	1.00	0.60	0.75	5
Black-and-White Fallacy	0.94	0.73	0.82	22
Causal Oversimplification	0.92	0.61	0.73	18
Doubt	0.80	0.68	0.74	66
Exaggeration/Minimisation	0.77	0.74	0.75	68
Flag-Waving	0.85	0.92	0.88	87
Loaded Language	0.79	0.91	0.85	325
Name Calling/Labeling	0.73	0.84	0.78	183
Repetition	0.71	0.56	0.63	145
Slogans	0.77	0.90	0.83	40
Thought-terminating Cliches	1.00	0.53	0.69	17
Whataboutism/Straw Men/Red Herring	0.76	0.45	0.57	29
Micro F1-Score			0.78	1063

Table 4.13: Classification Report Postprocessing Model

Class	Precision	Recall	F1-Score	Support
Appeal to Authority	0.91	0.71	0.80	14
Appeal to fear-prejudice	0.59	0.59	0.59	44
Bandwagon/Reductio ad hitlerum	0.75	0.60	0.67	5
Black-and-White Fallacy	0.94	0.68	0.79	22
Causal Oversimplification	0.80	0.44	0.57	18
Doubt	0.75	0.64	0.69	66
Exaggeration/Minimisation	0.74	0.74	0.74	68
Flag-Waving	0.83	0.86	0.85	87
Loaded Language	0.79	0.90	0.84	325
Name Calling/Labeling	0.75	0.83	0.79	183
Repetition	0.78	0.64	0.70	145
Slogans	0.76	0.80	0.78	40
Thought-terminating Cliches	0.75	0.53	0.62	17
Whataboutism/Straw Men/Red Herring	0.70	0.48	0.57	29
Micro F1-Score			0.77	1063

4.4.12 Adding Postprocessing

For Postprocessing, the solution was adopted by Chernyavskiy et al. [CIN20].

Model Changes

To increase the F1-Scores of *Repetition* and *Slogans*, the entire training dataset is searched for exact matches by removing punctuation, filtering out stopwords, and applying stemming. The Repetition class is assigned if the analyzed span matches at least two other spans. If only one match exists, the label is predicted with a threshold of at least 0.001. Also, if no match is found, the probability is set to zero unless the prediction probability of the Deep Learning model is at least 0.99. The *Slogans* technique is boosted by 0.5 if the span starts with a hashtag.

Results

When comparing the Classification Report of Meta Classification (see Table 4.12) and of Postprocessing (see Table 4.13) only the *Repetition* and *Name Calling/Labeling* techniques are predicted better. In contrast, all other classes are predicted worse, except *Whataboutism/Straw Men/Red Herring*, which has the same F1-Score in both reports. The overall Micro F1-Score drops by 0.01 to 0.77.

For Inference, having better predictions for eleven than two classes is desirable. Therefore Postprocessing will be discarded when analyzing the Analysis Dataset.

4.5 Evaluation

The chosen metrics for evaluation are Precision, Recall, F1-Score, and Support. These metrics are selected to provide a comprehensive view of the model's classification performance, considering each class's positive and negative predictions.

4.5.1 Evaluation of the Propaganda Detection Model

The Classification Report, presented in Table 4.12, shows the Meta Classification Model's performance on each propaganda technique.

The model achieved a Micro F1-Score of 0.78, indicating good performance in classifying the various propaganda techniques.

However, the performance varies across different classes. The model performs well on techniques such as *Loaded Language* (F1-Score: 0.85) and *Flag-Waving* (F1-Score: 0.88), while it struggles with classes like *Whataboutism/Straw Men/Red Herring* (F1-Score: 0.57) and *Repetition* (F1-Score: 0.63). The varying performance suggests that the model can be further improved to better recognize underperforming classes.

Analysis of Precision and Recall per Class

This chapter analyzes the distinct classes based on Precision and Recall. The thresholds set for the different clusters are relative.

High Precision and High Recall: High Precision means the model is good at accurately predicting positive instances, while high Recall means the model is good at capturing most of the positive instances.

- *Flag-Waving* (Precision: 0.85, Recall: 0.92)

Higher Precision but relatively Lower Recall: Again, higher Precision means the model accurately predicts positive instances, while lower Recall means the model is worse at capturing most positive instances.

- *Appeal to Authority* (Precision: 1.00, Recall: 0.71)
- *Bandwagon/Reductio ad hitlerum* (Precision: 1.00, Recall: 0.60)
- *Thought-terminating Cliches* (Precision: 1.00, Recall: 0.53)
- *Causal Oversimplification* (Precision: 0.92, Recall: 0.61)
- *Black-and-White Fallacy* (Precision: 0.94, Recall: 0.73)
- *Doubt* (Precision: 0.80, Recall: 0.68)
- *Exaggeration/Minimisation* (Precision: 0.77, Recall: 0.74)

Relatively Lower Precision but High Recall: Relatively lower Precision means the model is worse at accurately predicting positive instances, while high Recall means the model is good at capturing most positive instances.

- *Loaded Language* (Precision: 0.79, Recall: 0.91)
- *Slogans* (Precision: 0.77, Recall: 0.90)
- *Name Calling/Labeling* (Precision: 0.73, Recall: 0.84)

Relatively Lower Precision and very Low Recall: Relatively lower Precision means the model is worse at accurately predicting positive instances, while very low Recall means the model is terrible at capturing most positive instances.

- *Whataboutism/Straw Men/Red Herring* (Precision: 0.76, Recall: 0.45)
- *Appeal to fear-prejudice* (Precision: 0.73, Recall: 0.55)
- *Repetition* (Precision: 0.71, Recall: 0.56)

Analysis of F1-Scores per Class

Analyzing the F1-Score for each class helps understand how well the model is doing regarding both False Positives and False Negatives for each propaganda technique. The distinction into High, Moderate, and Low is made by applying the following thresholds:

- F1-Score > 0.8: High
- 0.6 <= F1-Score <= 0.8: Moderate
- F1-Score < 0.6: Low

Those thresholds are relative and only apply to the results of the Classification Report of the Meta Classification.

High F1-Score:

- *Flag-Waving*: 0.88
- *Loaded Language*: 0.85
- *Appeal to Authority*: 0.83
- *Slogans*: 0.83
- *Black-and-White Fallacy*: 0.82

These classes indicate that the model performs well in balancing Precision and Recall, leading to a high Micro F1-Score. High F1-Scores suggest that the model can effectively classify these propaganda techniques.

Moderate F1-Score:

- *Name Calling/Labeling*: 0.78
- *Exaggeration/Minimisation*: 0.75
- *Bandwagon/Reductio ad hitlerum*: 0.75
- *Doubt*: 0.74
- *Causal Oversimplification*: 0.73
- *Thought-terminating Cliches*: 0.69
- *Repetition*: 0.63
- *Appeal to fear-prejudice*: 0.62

The moderate F1-Scores for these classes indicate that the model's performance is neither particularly strong nor weak. The model's ability to classify these propaganda techniques is good, but there is room for improvement.

Low F1-Score:

- *Whataboutism/Straw Men/Red Herring*: 0.57

The low F1-Scores for this class show that the model has difficulty effectively classifying the *Whataboutism/Straw Men/Red Herring* technique. Low F1-Scores may arise from the model's inability to balance Precision and Recall, resulting in many False Positives, False Negatives, or both.

Comparison with the Baseline Model

A comparison between the *Baseline Model* Classification Report (see Table4.1) and the *Meta Classification Model's* Classification Report (see Table 4.12) reveals improvements in the Deep Learning system's performance in analyzing propaganda usage in news articles.

The *Meta Classification Model* (see Table 4.12) compared to the *Baseline Model* (see Table 4.1) is better any predicting any of the 14 propaganda techniques, as the following list shows:

- *Appeal to Authority*: The F1-Score increased from 0.0 to 0.83.
- *Appeal to fear-prejudice*: The F1-Score increased from 0.36 to 0.62.

- *Bandwagon/Reductio ad hitlerum*: The F1-Score increased from 0.0 to 0.75.
- *Black-and-White Fallacy*: The F1-Score increased from 0.0 to 0.82.
- *Causal Oversimplification*: The F1-Score increased from 0.29 to 0.73.
- *Doubt*: The F1-Score increased from 0.51 to 0.74.
- *Exaggeration/Minimisation*: The F1-Score increased from 0.52 to 0.75.
- *Flag-Waving*: The F1-Score increased from 0.68 to 0.88.
- *Loaded Language*: The F1-Score increased from 0.75 to 0.85.
- *Name Calling/Labeling*: The F1-Score increased from 0.69 to 0.78.
- *Repetition*: The F1-Score increased from 0.26 to 0.63.
- *Slogans*: The F1-Score increased from 0.31 to 0.83.
- *Thought-terminating Cliches*: The F1-Score increased from 0.1 to 0.69.
- *Whataboutism/Straw Men/Red Herring*: The F1-Score increased from 0.0 to 0.57.

Overall Micro F1-Score: The Micro F1-Score for the *Meta Classification Model* (0.78) indicates a notable improvement over the *Baseline Model* (0.54). This improvement demonstrates that the *Meta Classification Model* is more effective at classifying propaganda techniques in news articles.

4.5.2 Addressing Business Context

The following section resolves the identified Business Contexts in Chapter 4.1.

Business Context 1: Addressing Media Distrust

The first Business Context, which addresses media distrust, is effectively tackled by successfully identifying the various propaganda techniques in news articles.

For example, recognizing *Loaded Language* can facilitate a more objective understanding of information by revealing potential bias. *Exaggeration/Minimisation* can prevent recipients from distorting understanding if adequately recognized. *Appeal to Authority* can encourage the demand for evidence-based information. *Doubt* can be mitigated by organizations and individuals once recognized, working towards reestablishing trust. The misunderstanding of complex issues that arise from *Causal Oversimplification* can be avoided if the technique is identified, thereby promoting a more nuanced approach to complex topics.

Detecting and highlighting propaganda usage is instrumental in enhancing the credibility, transparency, and trustworthiness of media sources. Consequently, this empowers the general public to discern between reliable sources and those disseminating disinformation, fostering a more informed and engaged society.

This achievement is significant for stakeholders such as news organizations, journalists, media regulatory bodies, social media platforms, fact-checking organizations, educational institutions, and the general public. By revealing the underlying propaganda techniques, stakeholders can better comprehend and mitigate the factors contributing to media distrust, ultimately promoting a more transparent and reliable media landscape.

Business Context 2: Preserving Democratic Processes

In addressing the second Business Context, which focuses on preserving democratic processes, identifying propaganda techniques is essential. By successfully detecting and analyzing disinformation and propaganda content, the potential negative impact on democratic processes within the European Union and its Member States can be minimized.

Policymakers and stakeholders can better develop targeted countermeasures by understanding propaganda techniques and strategies. These countermeasures aid in combating disinformation and protect democratic processes, ensuring that the public's decisions are based on accurate information.

The distinct propaganda techniques can harm democratic processes by undermining the public's trust in democratic institutions, spreading disinformation, and fueling polarization: *Loaded Language*, *Name Calling/Labeling* can manipulate public sentiment by connecting strong emotions to specific subjects, parties, or individuals, ultimately leading to misinformed decision-making. *Repetition* and *Exaggeration/Minimization* can contribute to the distortion of facts, making it difficult for citizens to discern between truth and falsehood. *Appeal to fear and prejudice* can incite panic and further widen societal divides. In contrast, *Flag-waving* and *Bandwagon* tactics can exploit national sentiments or population groups to promote biased perspectives. *Causal Oversimplification* and *Thought-terminating Clichés* can stifle critical thinking and meaningful debate, discouraging a comprehensive understanding of complex issues. Techniques such as *Whataboutism/Reductio ad Hitlerum/Straw man* arguments can discredit opponents and obfuscate the truth.

Collectively, these propaganda techniques can potentially corrupt the foundations of democracy by fostering distrust, perpetuating disinformation, and weakening the ability of citizens to make informed decisions based on accurate information. By identifying these techniques, stakeholders such as European Union institutions, national governments, policymakers, political parties, electoral commissions, media organizations, civil society organizations, and the general public can engage in more informed discourse, contributing to a more robust and resilient democratic process.

Business Context 3: Restoring Trust in Institutions

In restoring trust in institutions, identifying and analyzing propaganda techniques is vital. By detecting propaganda in state-published media, awareness of manipulation-free communication can be raised. This increased awareness helps facilitate a more transparent and accountable media environment, ultimately restoring citizens' confidence in institutions.

Understanding the various propaganda techniques empowers stakeholders to recognize and address manipulation in media coverage, such as the attack on Ukraine. For instance, detecting *Loaded Language*, *Name-calling/Labeling* can help uncover attempts to manipulate public sentiment by connecting strong emotions to particular subjects or entities. Identifying *Repetition* and *Exaggeration/Minimization* can reveal efforts to distort facts and mislead the public. Recognizing *Appeal to fear or prejudice*, *Flag-waving*, and *Bandwagon* tactics can expose the exploitation of national sentiments or population groups to promote biased perspectives.

Moreover, stakeholders can encourage a more informed and meaningful discourse by discerning *Causal Oversimplification*, *Thought-terminating Clichés*, and other techniques that stifle critical thinking. This discourse contributes to a more robust and functional democratic system where citizens can make decisions based on accurate information. Consequently, trust in institutions can be restored as the public becomes more aware of manipulation-free communication and transparent media practices.

Business Context 4: Supporting Media Literacy

In conclusion, the ability to identify propaganda techniques through the Deep Learning system developed in this thesis significantly contributes to supporting media literacy. By providing a tool that detects these techniques in online news articles, stakeholders such as media literacy organizations, educational institutions, teachers, students, researchers, media watchdog groups, social media platforms, non-governmental organizations promoting media literacy, and the general public can enhance their understanding and critical evaluation of the information they consume. This, in turn, fosters a more discerning public better equipped to navigate the complex media landscape.

As citizens become more informed and develop the skills to recognize and identify propaganda techniques, they are less susceptible to manipulation and disinformation. Consequently, the media environment becomes more transparent, accountable, and trustworthy, promoting a well-informed society crucial for the functioning of a democratic system. Therefore, the Deep Learning system presented in this thesis is essential for empowering individuals and organizations to pursue media literacy and a more reliable and honest media landscape.

4.6 Deployment

The code used for building the Propaganda Detection Model was uploaded to GitHub[13].

[13] https://github.com/vitalijhein/propaganda-detection-thesis (last accessed: 29 May 2023)

CHAPTER 5

Results

This chapter aims to present the results of the three research questions.

5.1 In which way can the findings of the SemEval 2020 Task 11 [DSMBCW$^+$20] be combined, and to which degree in terms of F1-Score will this combination be able to compete with the results of the Technique Classification subtask?

As described in Chapter 4.5, the final Propaganda Detection Model performs with an 0.78 Micro F1-Score, after combining various approaches encountered in the challenge. Since the evaluation of test data was closed by Da San Martino et al. [DSMBCW$^+$20] after the challenge, scoring the model on test data is impossible.

Therefore, answering the Research Question is only possible on the scores of the development dataset the participants achieved. Table 5.1 lists the best five submissions, which submitted a solution on the development and test dataset. One participant only submitted a solution on the development dataset and did not submit a paper. Therefore, the submission will not be considered, even though it also ranked with 0.6717 Micro F1-Score [DSMBCW$^+$20] . The Propaganda Detection Model scores with a Micro F1-Score of around 0.78. Comparing it to the evaluation scores of the best five submissions shows superior performance in Micro F1-Score. The developed Propaganda Detection Model has no zero-predicted classes. Consequently, the combination of different techniques used by the participants does lead to even better results in terms of Micro F1-Score. In Chapter 6, the results are discussed in greater detail.

Participants	Micro F1-Score	Number of Zero-Classes
Jurkiewicz et al.	0.7046	0
Chernyavskiy et al.	0.6811	0
Li and Xiao	0.6783	0
Raj et al.	0.6717	0
Blaschke et al.	0.6689	1

Table 5.1: Top 5 Participants Development Dataset [DSMBCW+20]

5.2 What are the similarities and differences in using the different propaganda techniques when looking at Russian and American news articles?

5.2.1 Overall Analysis of the News Articles

Table 5.2 presents data on the number of articles, total sentences, detected propaganda, and detected percentage for various news publishers. The first publishers include ABC News, CBS News, CNN International, Fox News, and Politico. In contrast, the second group comprises News Front, Novye Izvestia, Russia Today, Sputnik News, and Tass News.

In summary, two distinct groups of publishers were analyzed. The first group, ABC News, CBS News, CNN News, Fox News, and Politico, had 12956 articles scraped, containing 567251 sentences. Among these, 114121 instances of propaganda were detected, making up an average of 20.11% of the sentences. With 2986 articles and 142043 sentences analyzed, CNN News showed the highest number of detected propaganda, 31041 (Detected Percentage: 21.85%). In contrast, despite having a larger volume of sentences with a total of 193472, ABC News showed fewer detected propaganda elements, with only 33362 (Detected Percentage: 17.24%).

The second group of publishers included News Front, Novye Izvestia, Russia Today, Sputnik News, and Tass News, from which 38457 articles were scraped, yielding 744649 sentences. In this group, 166061 instances of propaganda were detected, constituting 22.30% of the sentences on average. Russia Today, having the largest volume with 395496 total sentences analyzed, demonstrated the highest number of detected propaganda, 91005 (Detected Percentage: 23.01%). On the other hand, Tass News, with 111819 total sentences, showed the least detected propaganda instances, with only 21244 (Detected Percentage: 19%).

An in-depth discussion of the results can be found in Chapter 6. A small Propaganda Detection experiment on articles in Russian language reveals a 27.5% Detected propaganda rate in Russian domestic reporting and 13.12 propaganda techniques per analyzed articles.

Table 5.2: Overall Analysis of the Articles

Publisher	**Scraped Articles**	**Total Sentences**	**Detected Propaganda**	**Detected Percentage**	**Mean Sentences**	**Propaganda per Article**
ABC News	3998	193472	33362	17.24	48.39	8.34
CBS News	772	26346	5469	20.76	34.13	7.08
CNN News	2986	142043	31041	21.85	47.57	10.4
Fox News	4263	157643	34111	21.64	36.98	8.0
Politico	937	47747	10138	21.23	50.96	10.82
Summary	**12956**	**567251**	**114121**	**20.11%**	**43.78**	**8.81**
News Front	1432	30550	6145	20.11	21.33	4.29
Novye Izvestia	1378	47991	9950	20.73	34.83	7.22
Russia Today	20216	395496	91005	23.01	19.56	4.5
Sputnik	7784	158793	37717	23.75	20.4	4.85
Tass News	7647	111819	21244	19.0	14.62	2.78
Summary	**38457**	**744649**	**166061**	**22.30%**	**19.36**	**4.31**

5.2.2 Analysis of the American Articles

Table 5.3 lists the 14 propaganda techniques and their respective frequencies and percentages in the dataset. The technique used most frequently is *Thought-terminating Cliches*, comprising 8.64% (9864 instances) of the total distribution. *Thought-terminating Cliches* involve using phrases to halt argumentation or thinking on a topic.

The next most common technique is *Exaggeration/Minimisation*, representing 7.94% (9062 instances) of the data. This tactic involves overstating or understating elements of a claim or argument to make it appear more or less significant or impactful. *Bandwagon/Reductio ad hitlerum* and *Flag-Waving* are also significantly used, with frequencies of 7.51% (8571 instances) and 7.45% (8507 instances), respectively.

Loaded Language, an appeal to strong emotional implications to sway an audience's perception or opinion, was found in the dataset with 7.11% (8117 instances). *Doubt*, a technique designed to create uncertainty or skepticism, appears with 6.91% (7887 instances).

Appeal to Authority, *Appeal to fear-prejudice*, *Causal Oversimplification*, *Black-and-White Fallacy*, *Repetition*, *Slogans*, *Name Calling/Labeling*, and *Whataboutism/Straw Men/Red Herring* is also common, with *Name Calling/Labeling* appearing only with 6.17%, being the rarest propaganda technique in American news articles.

Table 5.3: Propaganda Techniques and Frequencies Detected in American Articles

Techniques	**Count**	**Percentage**
Appeal to Authority	7893	6.92%
Appeal to fear-prejudice	8271	7.25%
Bandwagon/Reductio ad hitlerum	8571	7.51%
Black-and-White Fallacy	7170	6.28%
Causal Oversimplification	7902	6.92%
Doubt	7887	6.91%
Exaggeration/Minimisation	9062	7.94%
Flag-Waving	8507	7.45%
Loaded Language	8117	7.11%
Name Calling/Labeling	7039	6.17%
Repetition	7496	6.57%
Slogans	8145	7.14%
Thought-terminating Cliches	9864	8.64%
Whataboutism/Straw Men/Red Herring	8197	7.18%

The variation in the usage of propaganda techniques across the five American news publishers - ABC News, CBS News, CNN News, Fox News, and Politico, help develop an understanding of their published propaganda. CBS News employs the technique of *Appeal to Authority* at a much higher rate, while Politico uses it half as much. The *Appeal* *to fear-prejudice* is predominantly used by ABC News, with CNN News implementing it the least. Examining *Bandwagon/Reductio ad hitlerum*, ABC News noticeably takes the lead, with CBS News utilizing it minimally. CBS News, Fox News, and Politico show an inclination toward the *Black-and-White Fallacy*. *Causal Oversimplification* is mainly used by Politico and rarest by CBS News and ABC News. *Doubt* is used by all publishers in a similar amount.

The *Exaggeration/Minimisation* strategy appears favored by CNN News, while Politico uses it less often. In the case of *Flag-Waving*, CNN News exhibits considerable usage, while CBS News shows low application. *Loaded Language* is most frequently found on Fox News, contrasting with CNN News and Politico's lesser usage. Conversely, *Name Calling/Labeling* seems to be employed substantially by Politico, while Fox News uses it the least. *Repetition* is prominently a CBS News technique, whereas ABC News relies less on it. Politico predominantly uses the *Slogans* strategy, while Fox News and CBS News use them less frequently. CBS News and Fox News substantially use *Thought-terminating Cliches*, with Politico at the lower end. Finally, CNN News and Politico utilize *Whataboutism/Straw Men/Red Herring* techniques most frequently, with ABC News showing the least utilization (see Table 5.4).

Table 5.4: Frequency of Propaganda Techniques in American News Outlets

Techniques	**ABC**	**CBS**	**CNN**	**Fox**	**Politico**
Appeal to Authority	6.44%	10.62%	6.97%	7.26%	5.17%
Appeal to fear-prejudice	10.00%	7.39%	5.19%	6.46%	7.06%
Bandwagon/Reductio ad hitlerum	8.98%	5.74%	6.77%	7.01%	7.57%
Black-and-White Fallacy	6.36%	6.98%	4.94%	6.99%	7.37%
Causal Oversimplification	5.92%	5.58%	7.46%	7.22%	8.31%
Doubt	6.74%	6.47%	7.48%	6.42%	7.63%
Exaggeration/Minimisation	7.74%	7.86%	8.98%	7.80%	5.95%
Flag-Waving	6.35%	5.41%	8.84%	7.83%	6.68%
Loaded Language	6.72%	6.91%	6.28%	8.48%	6.46%
Name Calling/Labeling	6.83%	6.60%	7.23%	3.98%	7.87%
Repetition	6.08%	7.52%	6.46%	7.04%	6.43%
Slogans	6.94%	8.10%	7.24%	6.42%	9.37%
Thought-terminating Cliches	9.29%	7.81%	7.65%	9.86%	5.93%
Whataboutism/Straw Men/ Red Herring	5.61%	7.00%	8.52%	7.23%	8.21%

5.2.3 Analysis of the Russian Articles

Table 5.5 outlines the frequency of the detected propaganda techniques found in Russian news articles. The most prevalent technique used is *Loaded Language*, (8.38% or 13776 cases), directly followed by *Appeal to Authority* (8.31% or 13667 cases). The next most frequent tactic is Black-and-White Fallacy (8.01% or 13171 instances). The three rarest detected propaganda techniques are *Repetition* (4.64% or 7622 cases), *Thought-terminating Cliches* (5.68% or 9336 cases), and *Slogans* (6.2% or 10198 cases).

Table 5.5: Propaganda Techniques and Frequencies Detected in Russian Articles

Labels	**Frequency**	**Percentage**
Appeal to Authority	13667	8.31%
Appeal to fear-prejudice	12420	7.56%
Bandwagon, Reductio ad hitlerum	11660	7.09%
Black-and-White Fallacy	13171	8.01%
Causal Oversimplification	11511	7.00%
Doubt	12689	7.72%
Exaggeration/Minimisation	12334	7.50%
Flag-Waving	12089	7.35%
Loaded Language	13776	8.38%
Name Calling/Labeling	11254	6.85%
Repetition	7622	4.64%
Slogans	10198	6.20%
Thought-terminating Cliches	9336	5.68%
Whataboutism, Straw Men, Red Herring	12659	7.70%

Table 5.6: Frequency of Propaganda Techniques in Russian News Outlets

Techniques	**News Front**	**Novye Izvestia**	**Russia Today**	**Sputnik**	**Tass News**
Appeal to Authority	6.88%	8.78%	8.70%	6.16%	7.13%
Appeal to fear-prejudice	4.82%	8.33%	6.46%	9.55%	8.32%
Bandwagon/Reductio ad hitlerum	8.71%	7.46%	5.66%	5.98%	7.06%
Black-and-White Fallacy	7.18%	9.25%	5.73%	7.91%	6.44%
Causal Oversimplification	7.90%	6.54%	7.02%	9.17%	7.31%
Doubt	6.32%	7.58%	8.85%	8.50%	6.07%
Exaggeration/Minimisation	8.97%	7.97%	5.85%	6.43%	7.73%
Flag-Waving	7.69%	7.52%	7.30%	6.49%	5.53%
Loaded Language	9.50%	7.16%	11.02%	7.97%	7.31%
Name Calling/Labeling	5.97%	7.01%	7.47%	5.50%	5.53%
Repetition	5.13%	4.20%	4.14%	7.33%	8.19%
Slogans	6.77%	5.88%	6.63%	5.02%	8.51%
Thought-terminating Cliches	3.92%	5.89%	5.43%	6.66%	8.17%
Whataboutism/Straw Men /Red Herring	10.23%	6.43%	9.72%	7.34%	6.70%

Next, Table 5.6 shows the relative propaganda technique distribution of the five Russian news outlets: News Front, Novye Izvestia, Russia Today, Sputnik, and Tass News.

The *Appeal to Authority* technique is most frequently used at Novye Izvestia and Russia Today, with Sputnik using it the least. *Appeal to fear-prejudice* sees its highest usage at Sputnik and Tass News, contrasting with News Front, which uses it minimally. The *Bandwagon/Reductio ad hitlerum* technique is most frequently used by News Front, with Russia Today using it the least. Novye Izvestia is noted for a dominant use of the *Black-and-White Fallacy*, whereas Russia Today and Tass News appear less inclined toward this technique. The *Causal Oversimplification* strategy is most heavily used by Sputnik, contrasting with Novye Izvestia, which uses it the least. *Doubt* is used significantly more by Russia Today and Sputnik, while Tass News and News Front use it least. *Exaggeration/Minimisation* is favored largely by News Front, with Russia Today implementing it the least.

Regarding *Flag-Waving*, News Front is also the top user, while Tass News uses it the least. The *Loaded Language* technique is most prevalent in Russia Today's content. However, Novye Izvestia utilizes it less frequently. *Name Calling/Labeling* is primarily used by Russia Today, contrasting with Sputnik and Tass News, which show minimal use of this technique. *Repetition* is more prominent in Tass News, while Russia Today and Novye Izvestia use it the least. Further, Tass News uses the *Slogans* technique more frequently, while Sputnik and Novye Izvestia uses it less often. Tass News also leads in using *Thought-terminating Cliches*, with News Front using this technique the least. Finally, *Whataboutism/Straw Men/Red Herring* techniques are often employed by News Front and Russia Today, while Novye Izvestia and Tass News show the least utilization.

5.2.4 Comparison of American and Russian Propaganda Usage

The analysis of propaganda techniques employed in American and Russian articles reveals similarities and differences. The analysis is based on Table 5.7.

The essential propaganda technique in Russian news articles is *Loaded Language*, in contrast to America, where the technique is the eighth detected technique. Looking at the *Appeal to Authority* technique in Russian articles, the technique is the second most used technique, while in American articles, the technique ranks only at place ten. Next, *Black-and-White Fallacy* is the third most used technique in Russian articles, while in American articles, it is one of the least important techniques, the second least detected technique. For *Doubt*, the fourth most used technique in Russian articles, the distribution is similar: In American articles, *Doubt* is only eleventh.

The technique *Whataboutism/Straw Men/Red Herring* is frequently detected by both countries' news publishers, ranking fifth in Russian and sixth in American articles. This is similar to *Appeal to fear-prejudice*, where the technique ranks sixth in Russian and fifth in American articles' propaganda encounters. *Exaggeration/Minimisation* is the seventh most detected technique in Russian articles, and is the most important in American ones. *Flag-Waving* behaves similarly: In Russian articles, eighth, in American ones third.

Nearly identical importance shows the *Causal Oversimplification* technique, tenth in Russian and ninth in American articles. Of less importance in both countries' articles is *Name Calling/Labeling*, being eleventh in Russian and least detected frequency in American articles.

A more considerable difference can be encountered in *Slogans*, rather not important in Russian (twelfth) versus seventh in American news coverage. A considerable difference in importance can be found in *Thought-terminating Cliches*. The technique is the most detected propaganda technique in American articles and the second least in Russian. Finally, *Repetition* being the least detected technique in Russian articles, and the twelfth detected in American ones, the technique does not appear to be a significant factor in both countries.

Table 5.7: Ranking of Russian and American Propaganda Usage

Rank	**Techniques in Russian articles**	**Techniques in American articles**
1	Loaded Language	Thought-terminating Cliches
2	Appeal to Authority	Exaggeration/Minimisation
3	Black-and-White Fallacy	Bandwagon/Reductio ad hitlerum
4	Doubt	Flag-Waving
5	Whataboutism,Straw Men, Red Herring	Appeal to fear-prejudice
6	Appeal to fear-prejudice	Whataboutism,Straw Men, Red Herring
7	Exaggeration/Minimisation	Slogans
8	Flag-Waving	Loaded Language
9	Bandwagon/Reductio ad hitlerum	Causal Oversimplification
10	Causal Oversimplification	Appeal to Authority
11	Name Calling/Labeling	Doubt
12	Slogans	Repetition
13	Thought-terminating Cliches	Black-and-White Fallacy
14	Repetition	Name Calling/Labeling

Analysis-based Conclusion on Russian Propaganda Usage in English-language Articles

The analyzed Russian articles often employ propaganda techniques to influence their audience effectively. Emotionally-charged language manipulates readers' feelings, pushing them toward specific viewpoints. These articles also frequently appeal to authority, validating their claims by citing perceived experts without providing further evidence. Moreover, complex matters are regularly presented as binary choices through the Black-and-White Fallacy, thus oversimplifying the discourse and potentially misleading readers. Techniques like Doubt undermine trust in individuals, institutions, or ideas that contradict the propagated narrative.

The articles also employ strategies like Whataboutism, Straw Men, and Red Herring to divert the focus away from the main issue or introduce irrelevant information to distract the audience. This diversion is often used to dodge criticism or avoid addressing complicated subjects. They also utilize fear and prejudice as tools to sway public sentiment. The articles can bolster support for their preferred narratives by stoking fear and panic about alternatives. Additionally, by exaggerating minor issues and minimizing major ones, they can control public perception and steer their audience's opinion in the desired direction.

Lastly, the analyzed Russian articles often exploit nationalistic sentiments or identities. By aligning their narrative with the readers' national pride or identity, they can rally support for their cause.

Analysis-based Conclusion on American Propaganda Usage

The analyzed American news articles frequently employ Thought-terminating cliches and Slogans to discourage critical thought and suppress meaningful debate. These cliches typically serve to terminate discussions by dissuading disagreement or promoting a simplified understanding. Furthermore, they utilize Exaggeration and Minimization techniques to manipulate the perceived importance of issues. These articles can direct public attention and shape opinion by amplifying some subjects and downplaying others. The Bandwagon and Reductio ad Hitlerum techniques are common; the former encourages readers to adopt mainstream perspectives, while the latter discredits ideas by associating them with opposing groups or concepts.

Flag-Waving is another prominent strategy, exploiting strong national sentiments or appealing to specific population groups to rally readers around a particular cause or idea. This technique often justifies or promotes certain narratives or initiatives.

Similarly to Russian articles, American ones use the Appeal to Fear and Prejudice, invoking fear and panic about potential alternatives and playing on existing biases to garner support for specific narratives.

Techniques like Whataboutism, Straw Men, and Red Herring are deployed to distract from the main argument. These involve accusing the critic of hypocrisy, misrepresenting an opponent's stance to refute the distorted position, or introducing irrelevant details to divert attention from the primary issue.

Slogans are also commonly used, offering short, catchy phrases that encapsulate a specific viewpoint. These Slogans, often laden with labeling and stereotyping, simplify complex issues and appeal to readers' emotions. Lastly, emotionally charged or Loaded Language is frequently used to provoke a potent response from readers, influencing their attitudes and behaviors.

5.3 How did the usage of different propaganda techniques change during the timeline of the Ukrainian War?

Unless stated otherwise, the following precognitions are relevant for any subsequent plots. Firstly, the analysis pertains to articles published from January 2022 through April 2023. The Russian attack on Ukraine commenced on 24 February 2022[1]. The comparisons are made relatively, given that more than three times as many Russian articles were collected. Normalization is done by counting the monthly encountering of propaganda elements and dividing by the yearly count. For 2023, where data is only available until April 2024, the data is projected for the remaining year based on the rate of the first four months.

5.3.1 Aggregated Distribution of Propaganda Technique

Figure 5.1 represents the overall distribution of encountered propaganda techniques in Russian and American news publisher articles. The plot displays the distribution of detected propaganda techniques per month. The left y-axis, which ranges from 0 to 0.5, illustrates this. The right y-axis shows the number of articles per month, highlighting the discrepancy in the volume of articles published by each country. The red line represents Russian articles, while the blue line represents American ones.

The most conspicuous spike in detected propaganda elements is evident in March 2022 from the American news publisher. This is also the only month American publishers produced nearly as many articles as their Russian counterparts throughout the entire time frame.

Before this spike, detected propaganda levels for both countries were comparatively low immediately following Russia's attack on Ukraine. Propaganda began to rise in February 2022, reaching its peak in March 2022. In April 2022, the number of published American articles nearly halved, while Russian articles decreased by a third.

In the following months, the number of published American articles and the detected propaganda techniques steadily declined until August 2022. However, starting in September 2022, the number of American articles and the detected propaganda began increasing again, peaking in February 2023, one year after the attack's onset. Afterward, the detected propaganda in American news articles started declining again.

The distribution of propaganda techniques in Russian articles paints an entirely different picture. While the spike in February 2022 was not as extreme as in the American articles, the usage of propaganda techniques maintained a consistent level. In the following months, the percentage of detected propaganda was generally higher than in the American articles, except for October 2022 and February 2023.

A deeper look at propaganda over time in Russian articles reveals the spike in detected propaganda techniques in February and March 2022. At the beginning of the attack,

[1] https://www.reuters.com/world/europe/events-leading-up-russias-invasion-ukraine-2022-02-28 (last accessed: 05 June 2023)

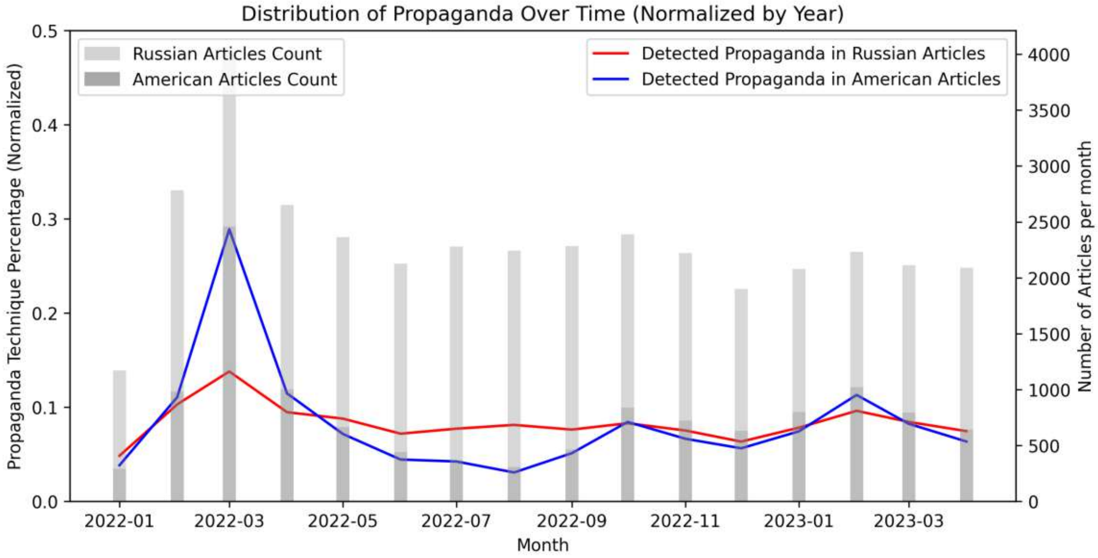


Figure 5.1: Overall Distribution of Propaganda Over Time

Novye Izvestia was the news publisher with the most detected propaganda techniques, followed by Sputnik News, Russia Today, News Front, and Tass. In the following months, detected propaganda decreased drastically in articles by Novye Izvestia, the publisher with the lowest encountered propaganda techniques from August to December 2022. Two spikes in January and March 2023 are returning it to the top propaganda distributor. While Sputnik spikes in February 2022 and 2023, the number of detected propaganda techniques stays stable over the months. Russia Today shows a distribution similar to Sputnik News, with a distinction in July 2022, where the publisher has the most detected propaganda. While News Front is inconspicuous during the attack, the publisher remains among the highest propaganda distributor between May 2022 and February 2023. It is important to note that for News Front, articles only from 24 February 2022 were collected, making the month unreliable. Finally, Tass News shows a relatively low propaganda count detected from January to July 2022. Still, after that, the news publisher quickly grows to one of the biggest propaganda publishers from August 2022 to April 2023. This could be due to restrictions in scraping data before July 2022 (see Figure 5.2).

Figure 5.3 now shows the distribution of detected propaganda per month in American articles. When looking at March 2022, the spike of published articles and detected propaganda indicates a big focus of the news publisher on the attack. The spike of detected propaganda in March 2022 is twice as high as in Russian articles, led by Fox News, Politico, CBS News, CNN News, and ABC News. Interestingly, the articles published by ABC News do not show many propaganda instances until September 2022, when the spike is the highest for them. It also keeps at the higher range for the following months. For American publishers, the attack on Ukraine is not a topic of immense interest, at least when looking at the decreasing number of published articles per month

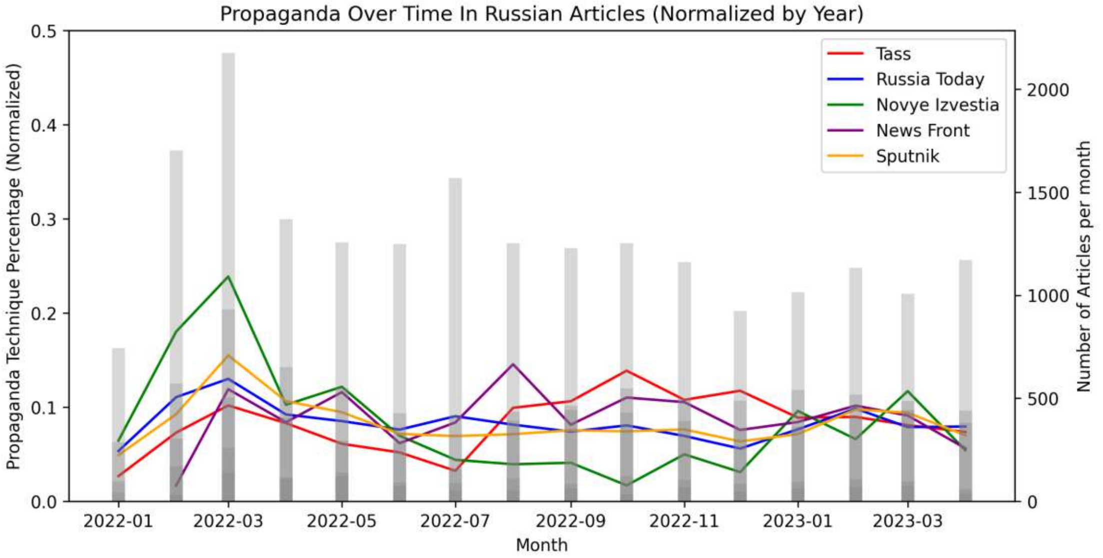


Figure 5.2: Propaganda Over Time in Russian Articles (Normalized by Year)

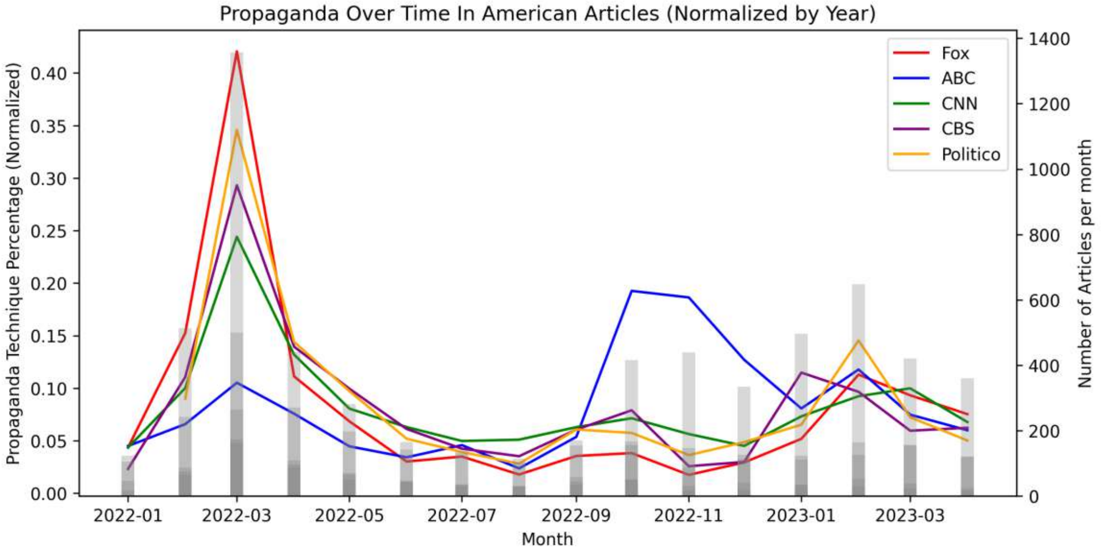


Figure 5.3: Propaganda Over Time in American Articles (Normalized by Year)

from April to August 2022. Then, the total published articles count doubles, while detected propaganda techniques stay low until January 2023, when detected propaganda again grows and spikes in February 2023 for all publishers.

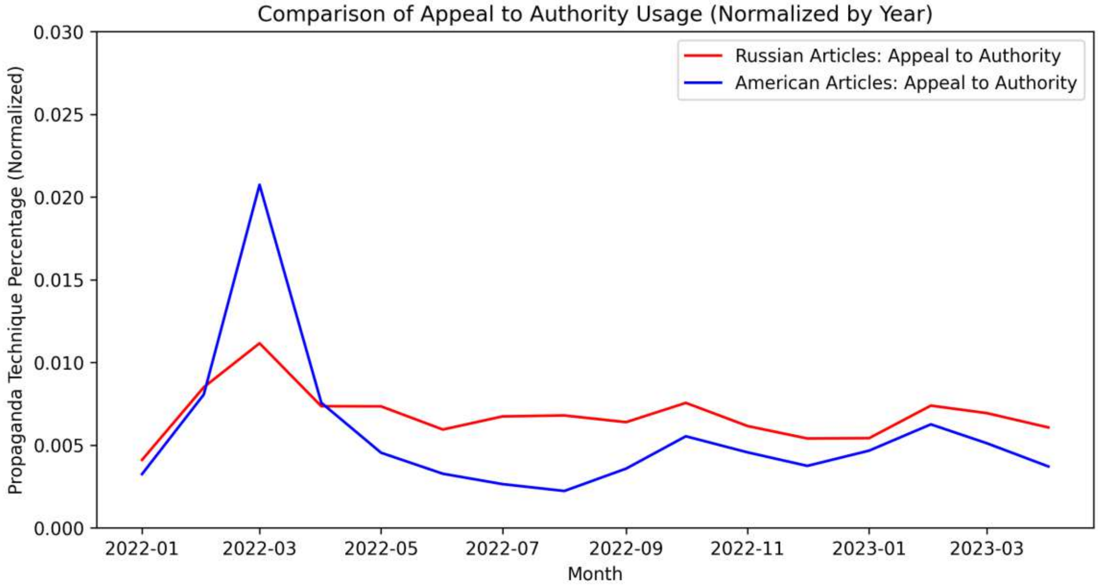


Figure 5.4: Comparison of Appeal to Authority Usage

5.3.2 Monthly Aggregated Distribution per Propaganda Technique

This chapter hosts the plots for the 14 propaganda techniques. Every propaganda technique is analyzed beginning from January 2022 to April 2023. The y-axis ranges from 0 to 0.03 for each plot. The red line always corresponds to the detected propaganda in Russian articles, while the blue line is for American articles. Figure 5.4 shows the frequency of this technique peaks in March 2022 for Russian and American Articles. It then decreases in April 2022 but remains relatively stable until August 2022 for Russian articles. There is a slight rise in October 2022, followed by a decrease towards the end of the year. In 2023, the frequency of this technique shows an increase in February, then a slight decrease through April. The American articles frequency decreases substantially in April 2022 and remains relatively low through the summer. Again there is a slight peak in October 2022, followed by a decrease towards the end of the year. In 2023, there is a rise in the frequency of this technique in January and February, followed by a decrease in March and April.

The distribution of the *Appeal to fear-prejudice* technique can be seen in Figure 5.5. Russian articles show a significant peak in the usage of this technique in March 2022. The frequency then decreases in April 2022 but stays relatively high compared to the start of the year. It fluctuates throughout the rest of the year, with its lowest frequency in June 2022. Starting in 2023, there is a rise in January and February, followed by a decrease in March and April. American articles also show a peak on 22 March. After April 2022, the frequency stays relatively low but has a small peak in October 2022. January and February 2023 show a rise in frequencies, with a subsequent decrease.

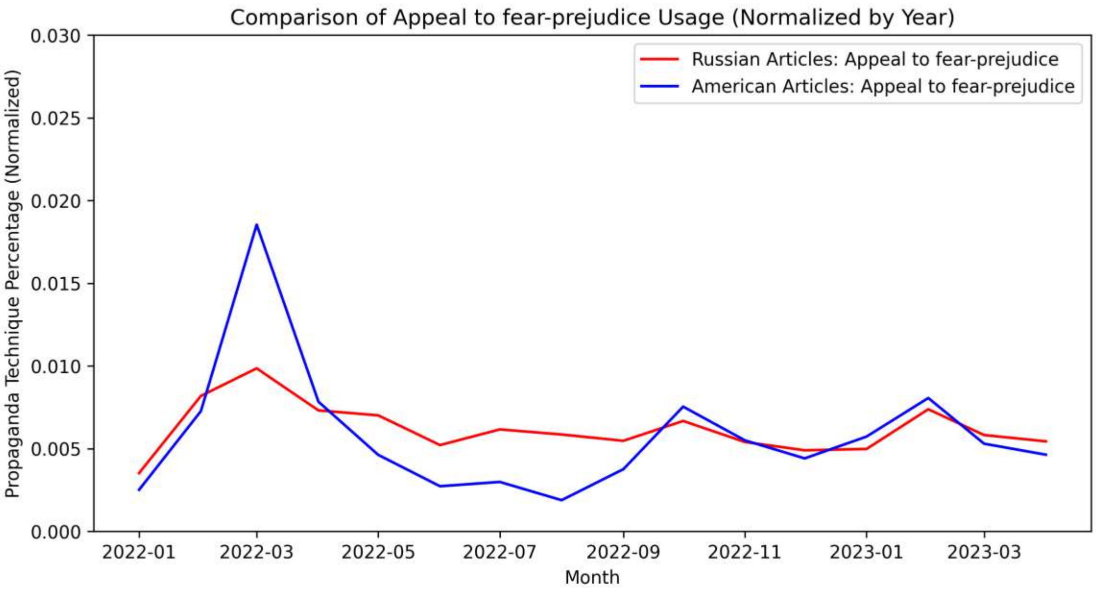


Figure 5.5: Comparison of Appeal to fear-prejudice

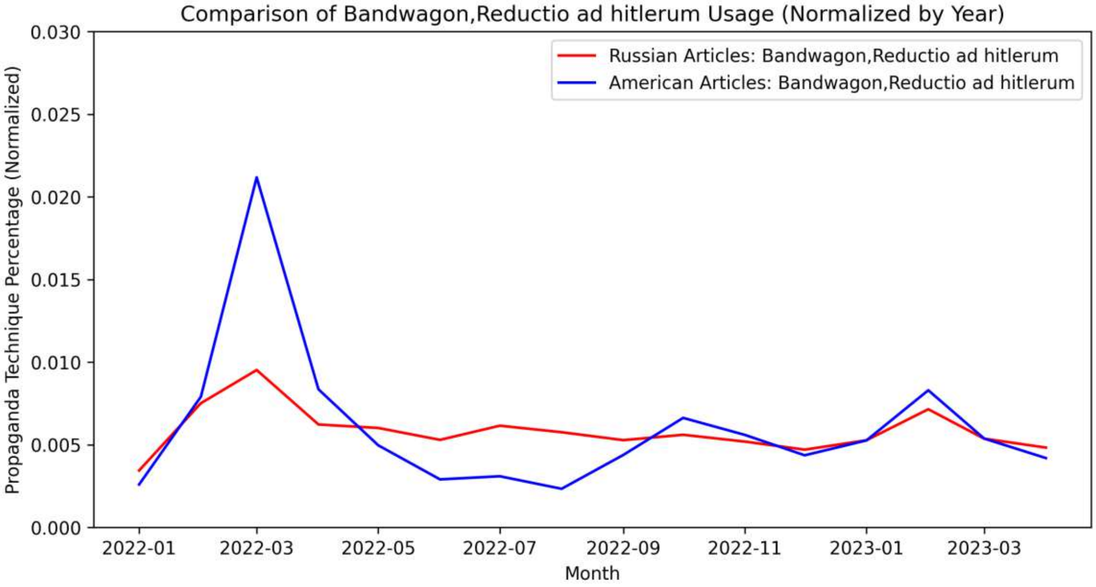


Figure 5.6: Comparison of Bandwagon/Reductio ad hitlerum Usage

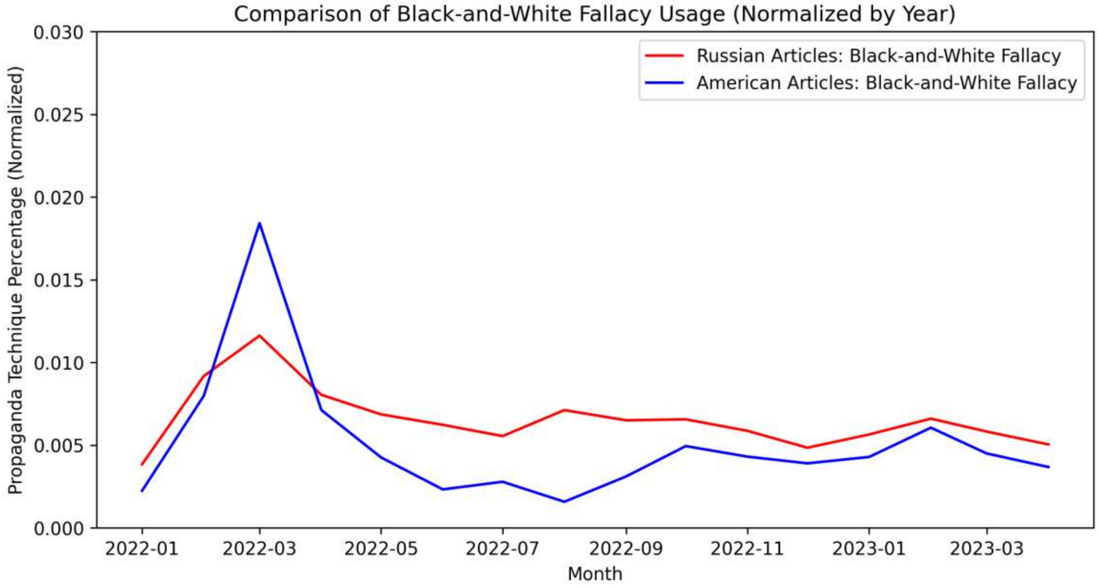


Figure 5.7: Comparison of Black-and-White Fallacy Usage

The propaganda technique *Bandwagon/Reductio ad hitlerum* (see Figure 5.6) was mainly utilized in March 2022, but the peak is significantly higher in American articles. After May 2022, the frequency drops below the Russian articles' frequency, surpassing it only in October 2022 and February 2023. In November 2022, January, and March 2023, both Russian and American articles have a similar technique frequency. The Russian frequency is stable throughout the months, with smaller peaks in July, October 2022, and February 2023.

The technique *Black-and-White Fallacy* (see Figure 5.7) shows an interesting tendency: Except for March 2022, where American technique frequency peaks, Russian article frequency is higher for every remaining month in 2022 and 2023. After March 2022, the frequency of Russian articles decreases but keeps peaking in August, October 2022, and January 2023. American articles' frequency decreases to a minimum in May and August 2022. After August 2022, the technique frequency keeps growing until February 2023. For both countries, the technique frequency decreases in March and April 2023.

Figure 5.8 shows the frequency of the *Causal Oversimplification* propaganda technique. For the Russian articles, the frequency reached its peak in March 2022. Following this peak, there is a decrease in April 2022, but the frequency remains relatively stable, with slight fluctuations through the rest of 2022. From January 2023 to February 2023, there is a slight increase in usage, which then decreases through April 2023. In American articles, the usage of this technique also peaks in March 2022, showing a significantly higher frequency than in Russian articles. The frequency then decreases substantially from April 2022 to August 2022. It peaks again in October 2022 and January 2022.

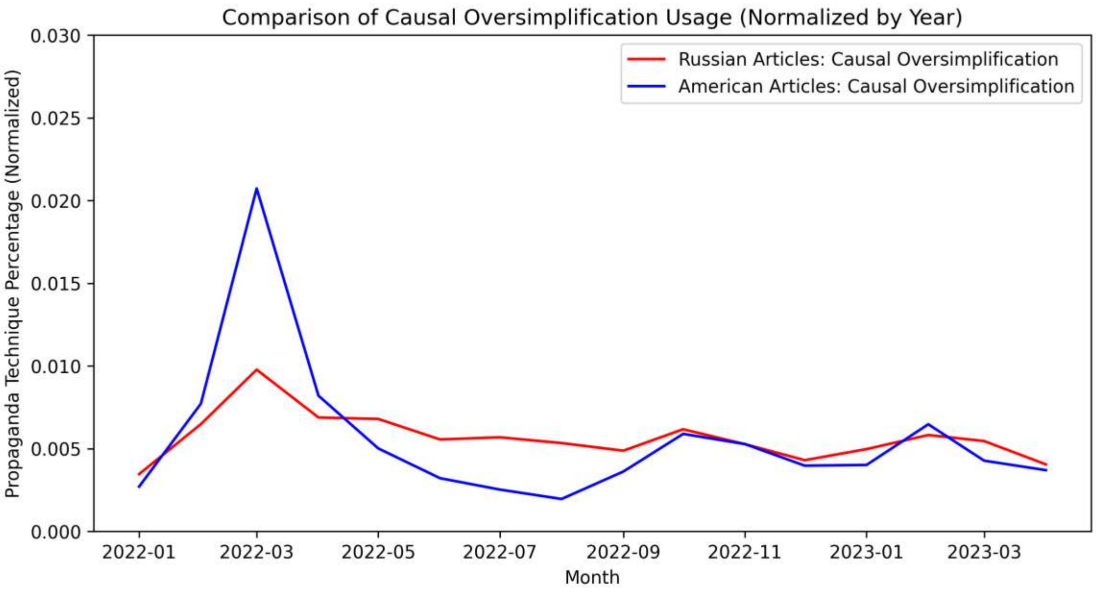


Figure 5.8: Comparison of Causal Oversimplification Usage

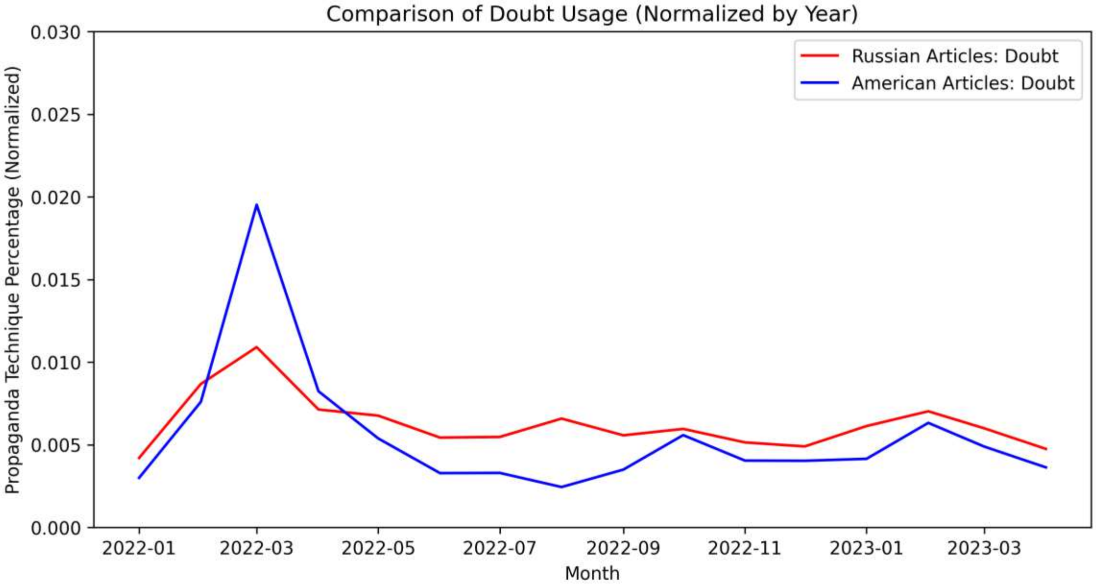


Figure 5.9: Comparison of Doubt Usage

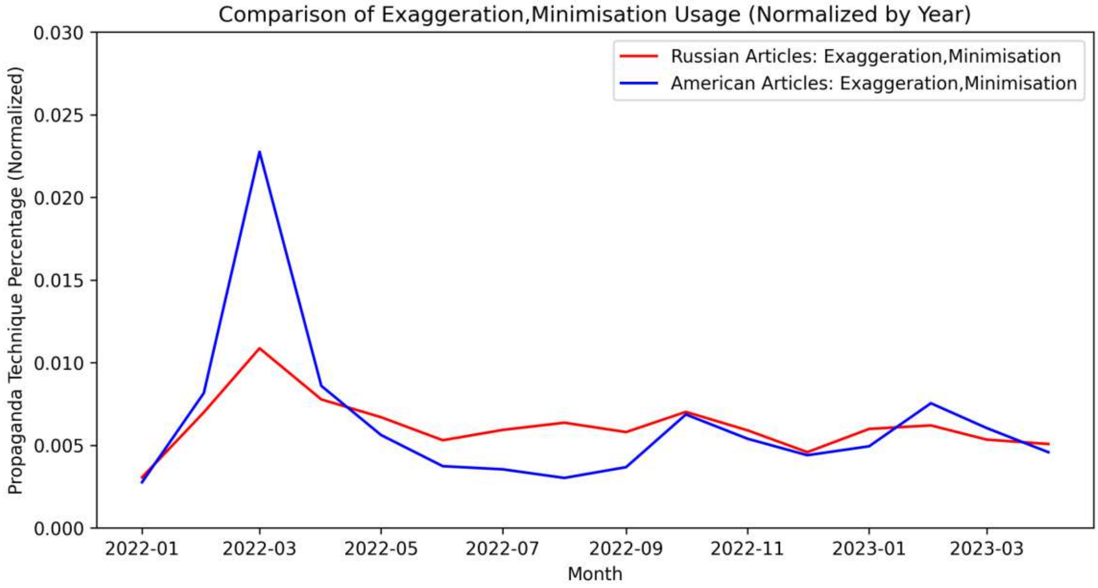


Figure 5.10: Comparison of Exaggeration/Minimisation Usage

Next, the propaganda technique *Doubt* is shown in Figure 5.9. For the Russian articles, there is a significant peak in usage around March 2022, with a frequency slightly above 0.01. The usage frequency decreases afterward, but smaller peaks are observed in August 2022 and February 2023. For the American articles, the usage frequency peaks around March 2022, reaching just under 0.02. This is the highest frequency observed in the plot. There is a significant drop in usage following this peak, and the frequency remains relatively low, with minor peaks in October 2022 and February 2023. After April 2022, American articles always show less usage of *Doubt* than Russian ones.

Figure 5.10 represents the frequency of the *Exaggeration/Minimisation* propaganda technique usage. Again for both countries, there is a peak in March 2022, where the American peak is more than twice higher. The usage decreases from April 2022 until August 2022 and again peaks in October 2022 and February 2023. Russian usage is higher from May 2022 to September 2022, November 2022, January 2022, and January 2023. The remaining months show more usage in American news articles.

Figure 5.11 shows the *Flag-Waving* technique frequency. There is a noticeable peak in usage around March 2022 for Russian and American articles, with a frequency slightly below 0.01 for Russian and above 0.02 for American articles. From May 2022 to August 2022, for American articles, there was a significant drop in usage following this peak. The other peaks are around October 2022 and February 2023. For Russian articles, the usage frequency appears to decrease after the peak in March 2022, with minor fluctuations throughout the rest of 2022 and 2023.

Figure 5.12 shows that Russian and American articles fluctuate using the *Loaded Language*

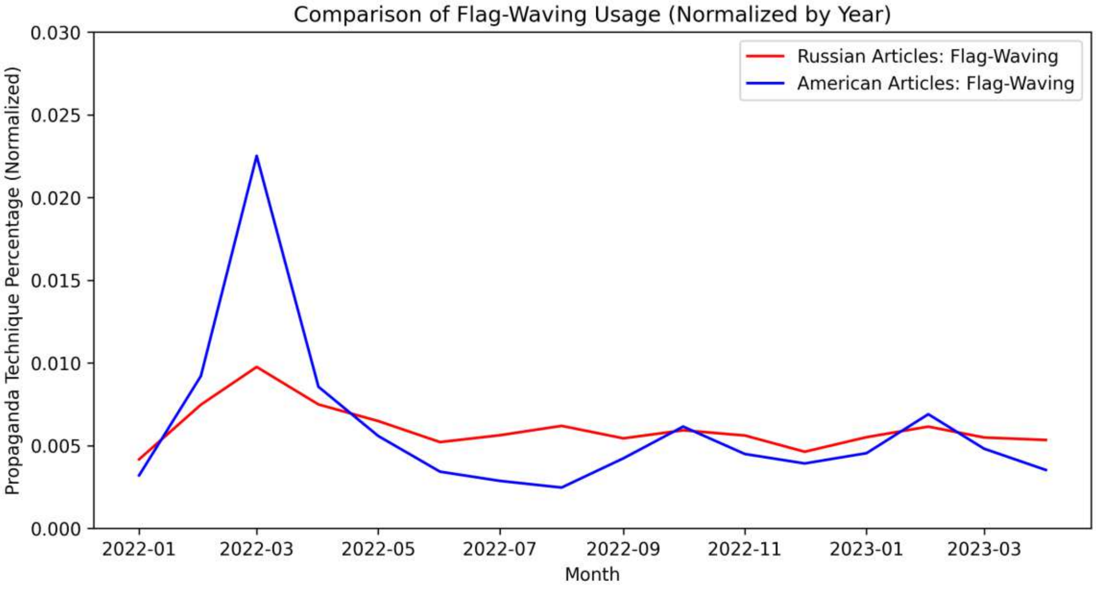


Figure 5.11: Comparison of Flag-Waving Usage

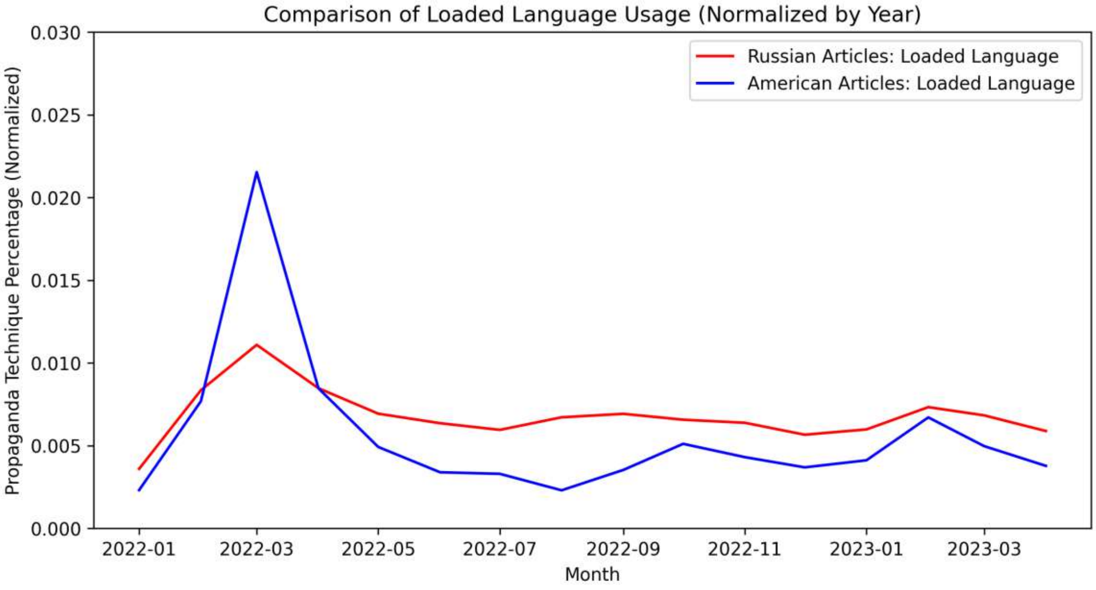


Figure 5.12: Comparison of Loaded Language Usage

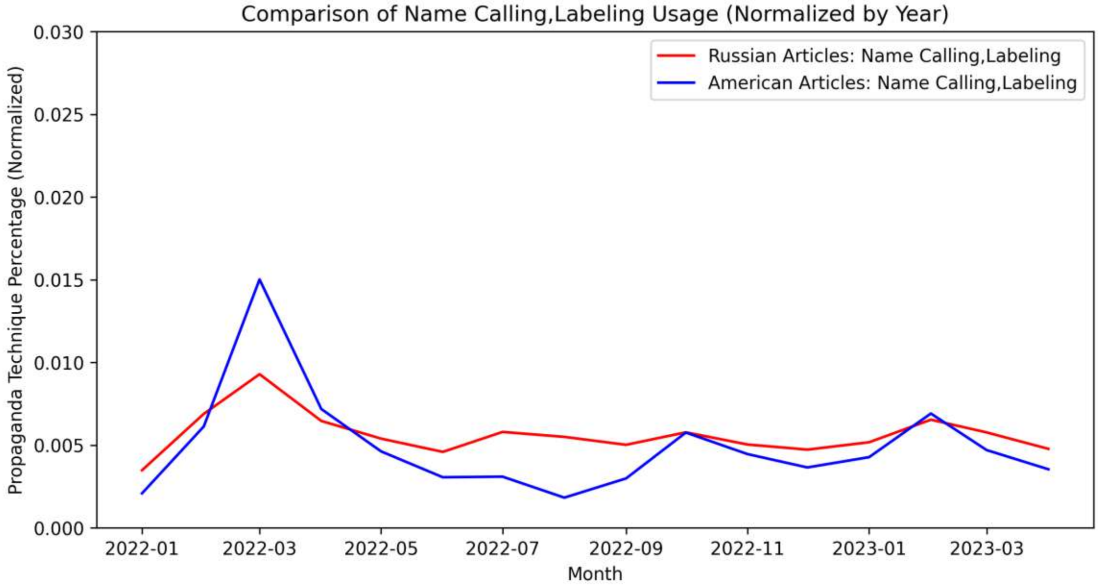


Figure 5.13: Comparison of Name Calling/Labelling Usage

propaganda technique from January 2022 to April 2023. Russian articles show noticeable peaks in usage around March 2022 and February 2023, both slightly above 0.01. The usage frequency generally decreases over the observed period, with minor fluctuations. For the American articles, the usage frequency appears to peak around March 2022, reaching just above 0.02. There is a significant drop in usage following this peak, and the frequency remains relatively stable for the rest of 2022. In October there is a slight increase, followed by a decrease again. In 2023, there is a slight increase in usage in February, followed by a decrease. When comparing the two countries, American articles show a higher peak in using the *Loaded Language* propaganda technique (in March 2022), but their usage frequency decreases significantly afterward. On the other hand, Russian articles exhibit a more consistent usage of this technique over time, with minor fluctuations and a general downward trend.

The *Name Calling/Labeling* technique (see Figure 5.13) shows a significant peak in the usage of this technique in March 2022 for both countries. However, the peak for American articles shows a significantly higher frequency than the Russian articles. The Russian articles' frequency is lower in February 2022, March 2022, and February 2023. In the remaining months, the frequency of usage is higher for Russian articles. American articles show a significant peak in October 2022 and February 2023.

In Russian articles, the *Repetition* technique (see Figure 5.14) initially increases from January 2022, reaching a peak in March 2022. The frequency decreases in April 2022 and continues to fluctuate over the next months, with another small peak in October 2022. There is a substantial drop in December 2022 and January 2023. It increases again in February 2023 and then decreases again until April 2023. American articles show a

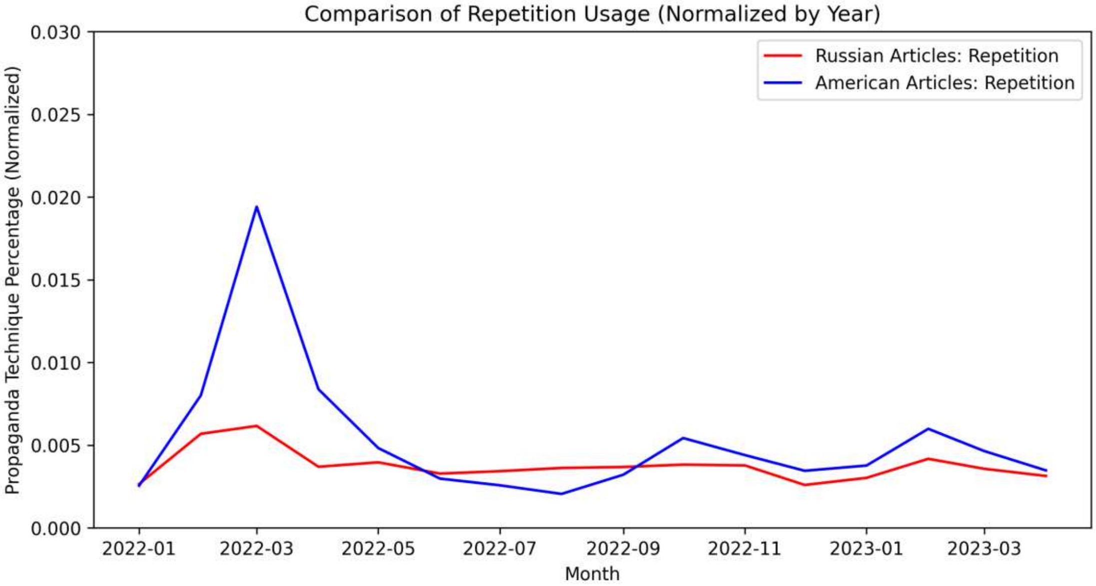


Figure 5.14: Comparison of Repetition Usage

significant peak in the usage of this technique in March 2022, which is more pronounced than the Russian articles. The frequency decreases substantially after April 2022 and keeps decreasing, reaching another peak in October 2022, similar to the Russian trend. From January 2023, there is a rise in usage, followed by a decline through April 2023.

Looking at *Slogans* (see Figure 5.15) again, March 2022 shows the highest peak for both American and Russian articles, but the American frequency is more than twice higher. Russian technique frequency then decrease but stays relatively stable, while American frequency decreases until August 2022, when it reaches its lowest point. In September and October 2022, the usage frequency increases again for both. After that, the frequencies are similar, with slight differences from February to April 2023.

Thought-terminating Cliches being the most used propaganda technique in American articles, Figure 5.16 shows this exactly. American articles utilize the technique more often in all months except from June 2022 to September 2022—the usage frequency peaks significantly around March 2022, October 2022, and February 2023. For the Russian articles, there is a significant peak in usage around March 2022. The usage frequency decreases afterward.

For Russian articles, the *Whataboutism/Straw Men/Red Herring* technique in Figure 5.17 shows a noticeable peak in usage around March 2022. Russian usage of *Whataboutism/Straw Men/Red Herring* appears to be higher in the following months, except for October 2022 and January 202, where American articles show two spikes in usage. In March and April 2022, American publishers also utilized the technique predominantly. Comparing the two countries, American articles show a higher peak

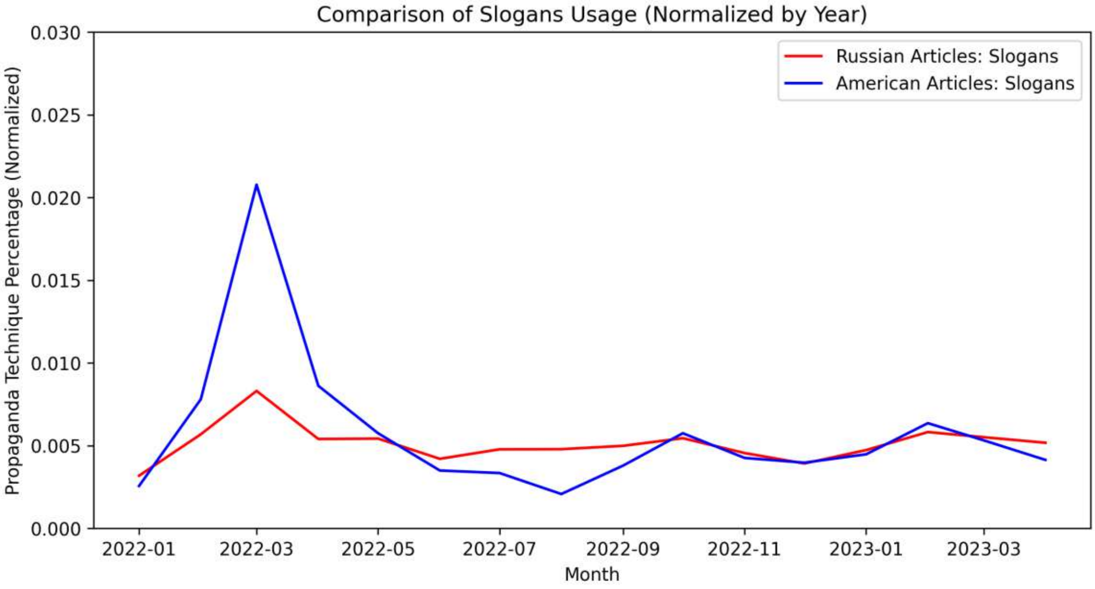


Figure 5.15: Comparison of Slogans Usage

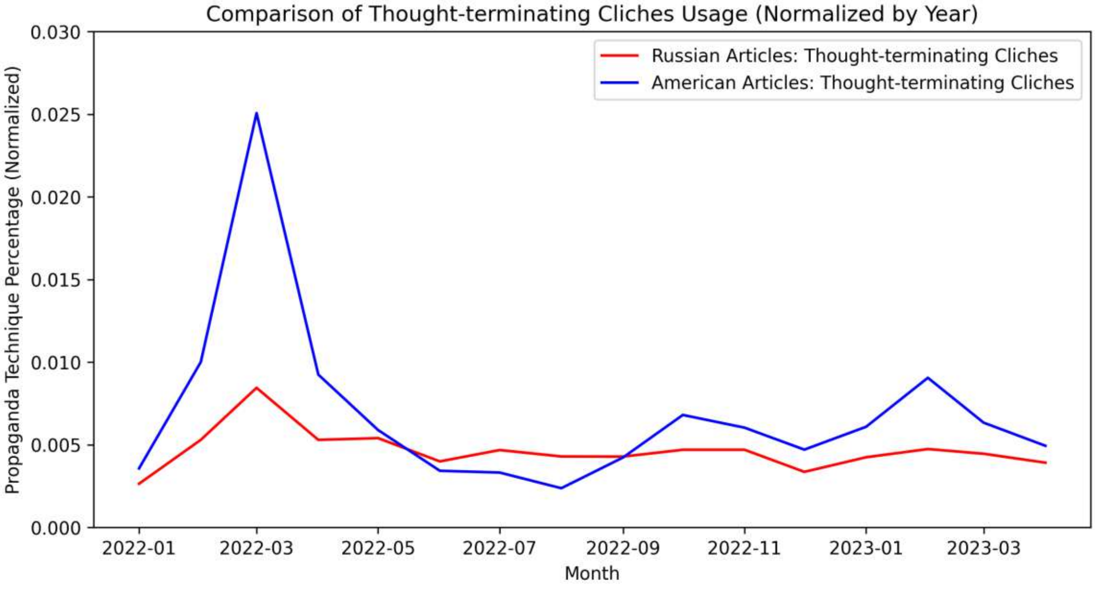


Figure 5.16: Comparison of Thought-terminating Cliches Usage

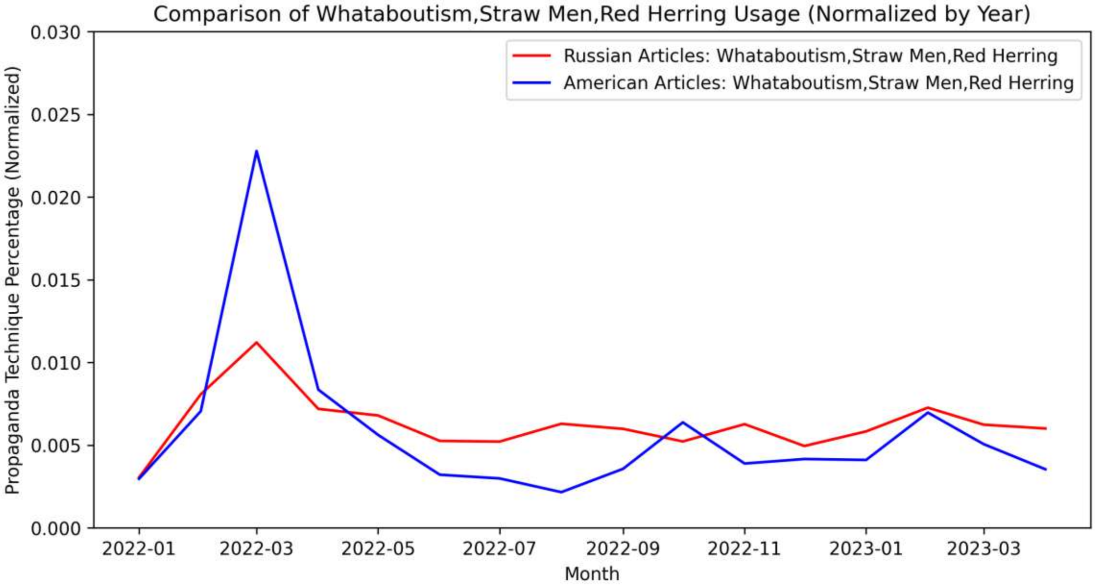


Figure 5.17: Comparison of Whataboutism/Straw Men/Red Herring Usage

in using the *Whataboutism/Straw Men/Red Herring* propaganda techniques (in March 2022), but their usage frequency decreases significantly afterward. On the other hand, Russian articles exhibit a more consistent usage of these techniques over time, with minor fluctuations and a general downward trend.

CHAPTER 6

Discussion

In this chapter, the efforts and issues to construct an optimized model for Propaganda Detection are discussed. Further, limitations and further experiments are thematized.

6.0.1 Combining Deep Learning Approaches

The first Research Question primarily concerns connecting knowledge from different Deep Learning areas to build a superior model.

Most successful participants worked with Transformer-based Language Models, justifying the incorporation of Transfer Learning and Transformer Models. Jurkiewicz et al. [JBKG20], who placed first, and Morio et al. [MMOM20], who placed third, both used a setup with multiple models and Hyperparameter Optimization, thereby forming the starting point for optimizing the Baseline Model. Further, four of the five top submissions used the surrounding context of a propaganda span, indicating its importance in the Propaganda Detection Model.

Next, the Averaged Embedding and Span Length, as described by Chernyavskiy et al. [CIN20], was leveraged, leading to the best-performing model before Ensemble Learning. Evaluating the Literature Analysis, Cost-Sensitive-Learning was referenced by multiple participants like Jurkiewicz et al. [JBKG20], Grigorev and Ivanov [GI20], and Li and Xiao [LX20]. While the participants derived positive results, different attempts to implement it resulted in bad-performing models.

Over- and Undersampling seemed to be reasonable attempts to equalize the difference between Minority and Majority Classes, as proposed by Jiang et al. [JGM20] and Grigorev and Ivanov [GI20]. Furthermore, even though Minority Classes were predicted better by the Propaganda Detection Model, the overall Micro F1-Score decreased substantially.

Upon analyzing the Averaged Embedding Model, substantial improvements were made by adding new features. In their submission, the second-placed Morio et al. [MMOM20] used Part-of-Speech Tags and Named Entity Tags. Creating a Part-of-Speech Embedding and adding it to the Averaged Embedding Model did not help increase the Micro F1-Score. Testing the combination of a Part-of-Speech Embedding with the plain RoBERTa Embedding also did not help.

After previous attempts resulted in no significant improvements, the decision was made to use existing models, such as the optimized Baseline and the Averaged Embedding, and train them using different Transformer-based Language Models. Morio et al. [MMOM20] used this approach to train multiple models and combine them during Ensemble Learning. The existing model architectures were used to train Language Models like BERT and OPT. The resulting weak learners combined during Ensemble Learning led to the superior Propaganda Detection Model.

Lastly, the Postprocessing approach described by Chernyavskiy et al. [CIN20] was tested to improve the predictions for the *Repetition* and *Slogans* classes. Unfortunately, this process improved the performance of the weak learners but not of the Propaganda Detection Model. On the contrary, the Micro F1-Score decreased.

While many strategies were proposed, those implemented by the top five submissions were preferred. Nevertheless, some exciting yet time- and resource-consuming ideas were not implemented.

6.0.2 Dataset and Model Limitations

The following dataset and model limitations apply only to weak learners before using Ensemble Learning. The issues described here are not relevant after that.

The models were trained on articles from American publishers only, possibly resulting in greater robustness with the American writing style. Furthermore, the RoBERTa and BERT models, released in 2019, are not updated with events post-release, potentially reducing their stability in recognizing contemporary contexts. The same problem occurs with the Propaganda Technique Corpus. The dataset was released in 2019 and not updated afterward. Hence, the Propaganda Detection Model was fine-tuned on articles from before 2019, potentially favoring the recognition of Majority Classes over Minority Classes. As news articles emphasize current events, the model might struggle to apply its learning from historical contexts to current ones.

Despite the dataset's robust base, model enhancement requires more labeled data, particularly given the challenges faced by Minority Classes before Ensemble Learning. Due to the highly imbalanced Propaganda Technique Corpus, weak learners trained on this dataset tend to replicate technique distribution within unseen data. For example, the propaganda technique *Loaded Language* made up about 50% of occurrences across all ten analyzed news publishers while also being the most prevalent technique in the Propaganda Technique Corpus. The dominance of this class becomes apparent when considering the span length and some examples, which often consist of single words. Consequently, over-descriptive, loaded adjectives may be employed more frequently to create engaging articles. In contrast, a sentence containing a technique like *Whataboutism* requires a more complex sentence structure, beginning with an argument needed to be relativized, followed by a sentence that can relativize the previous argument.

Regarding the Propaganda Technique Corpus, it is uncertain whether its class distribution accurately reflects the distribution of propaganda techniques in news articles over time or merely captures a snapshot of American news articles published before 2019. Therefore, it is unclear if a more balanced distribution of propaganda techniques could lead to better prediction of Minority Classes at the expense of distorting real-world propaganda usage. Similarly, it is unknown if the current class imbalance should be preserved while increasing the number of examples according to the class distribution. Finally, it is also not said that including more recent news articles would help improve Propaganda Detection Models, even though the suggestion was made previously to enhance predictions of Minority Classes.

Examining the Analysis Dataset reveals potential differences in publishing styles that could further complicate matters. For instance, Tass News publishes shorter statements with more formal language. In contrast, Fox News tended to write lengthy articles with descriptive language and ample room for interpretation and author opinions. The limited availability of English-language Russian news publishers contributed to these discrepancies. Furthermore, the lack of a labeled propaganda dataset in the Russian language hinders the analysis of articles written in Russian.

Lastly, the Span Identification model proposed by Chernyavskiy et al. [CIN20] may pose a challenge, as the model's prediction errors can carry over to the Technique Classification model. This issue arises because potentially incorrect spans served as inputs for classification, increasing the likelihood of erroneous predictions. Since the Propaganda Detection Model was trained on propaganda spans, it struggled to classify techniques accurately by inputting the whole propaganda sentence. Additionally, both models were trained on the same imbalanced dataset, potentially leading to a bias toward Majority Classes during Span Identification.

6.0.3 The Issue with Micro F1-Score in the SemEval Challenge

The Micro F1-Score, though seemingly effective, encounters a crucial issue: its enhancement can be achieved if the model disregards Minority Classes. Consequently, the Micro F1-Score increased due to its consistent prediction of Majority Classes while becoming entirely unreliable for Minority Classes. In the context of the SemEval challenge, participants could potentially attain high Micro F1-Scores by sacrificing Minority Classes. Although this strategy might secure victory within the competition, it raises doubts about its practicality in the broader scope of Propaganda Detection. As a result, for future contests on the Propaganda Technique Corpus, challenge organizers should incorporate an additional criterion: "Highest Micro F1-Score with every class classified".

6.0.4 Computational Power and Resources

Using BERT and RoBERTa, developed in 2018 and 2019, respectively, as the primary models for building the Propaganda Detection Model might raise questions. First, when the development process began, Artificial Intelligence was a specialized topic in public discourse. Public discussions focused on Large Language Models like GPT-3 and their potential future applications. The release of ChatGPT[1], a fine-tuned version of GPT-3 capable of interacting in a chat environment, spurred excitement around AI and strengthened scientific research. Subsequently, GPT-4 was introduced, marking the most successful Transformer Model ever created. Also, institutions like Meta unveiled OPT, a remodeled version of GPT-3, designed to demonstrate model construction and support scientific research transparently. Meta also released LLaMA, a GPT-4 clone with similar aims as OPT.

While OPT and LLaMA are freely available, they cannot be run on ordinary hardware. Training and evaluating such models require dedicated hardware, thus highlighting one of the thesis's significant limitations: inadequate resources and computational power. To train models like RoBERTa-large, an environment like Google Colab[2] with access to a GPU is required. Although GPU access facilitated rapid training of even large models like RoBERTa with 125 million parameters, working with higher-parameter models like OPT with 1.3 billion parameters was nearly impossible. After reducing the batch size to four samples per batch, the available GPUs could barely train the OPT model with 1.3 billion parameters, but training time remained an issue. One training epoch took approximately 30 minutes, contrasting the 3 minutes for RoBERTa-base and 9 minutes for RoBERTa-large. Even after training the OPT-1.3B model for three 30-minute epochs, the results were worse than RoBERTa-large. While model adjustments and Hyperparameter Optimization might have been helpful, the associated costs were prohibitive. Around €300 was spent renting computational power in the cloud environment. While training the model would have been less expensive, developing and testing new hypotheses also demanded resources, leading to considerable costs.

As a result, the decision was made to use smaller yet older models. Fine-tuning large-scale models proved too costly, time-consuming, and outside the thesis's possibilities. Nonetheless, when large models were tested during this thesis, the results did not differ dramatically between RoBERTa-base or -large implementations. In the later process, the OPT model with 1.3 billion parameters was still leveraged during Ensemble Learning and improved the Micro F1-Score from 0.75 to 0.78.

According to Sam Altman, CEO of OpenAI, the era of continuously larger Language Models may be ending [Mil23]. Interestingly, the company's next focus will not be developing larger model architectures with more parameters as previously. This aligns with the thesis that most changes in model architecture during the propaganda detector development were not helpful, and the most remarkable improvement in Propaganda

[1] https://chat.openai.com (last accessed: 29 May 2023)

[2] https://colab.research.google.com (last accessed: 29 May 2023)

Detection would come from creating a better, larger, and more balanced Propaganda Technique Corpus dataset.

6.0.5 Analyzing Russian-language Articles from Russia Today

Analysis of Russian and American news articles revealed a significant difference in length. On average, American articles contain 43.78 sentences, while Russian articles are more than twice as short, with only 19.36 sentences each.

The different news sources made it evident that the articles obtained from the countries varied in their style. For instance, Tass News issues brief press releases focused solely on delivering information. In contrast, Fox News typically publishes longer articles that incorporate entertainment elements and often include the author's perspective.

After searching for another Russian news source, that would show similar characteristics as American news publishers, an interesting observation was made after checking the Russian-language version of Russia Today with its corresponding English-language version. The Russian version would host an extra category named *Opinion*, where Russian authors would write lengthy articles on current topics incorporating their personal opinion.

Since no better alternative was found, those articles were downloaded. In the timespan from January 2022 to April 2023, around 723 articles were obtained this way. Next, the articles were translated from Russian to English with the Marian model [JDGD+18]. The translated articles were handled in the same manner as described in Chapter 4. After running the Propaganda Detection Model on these translated articles, the findings show an average propaganda frequency of 27.50% for those articles. The average sentence count in these translated articles (47.71%) is comparable to that of CNN News (47.71%), ABC News (48.39%), and Politico (50.96%). The actualized overview of Table 5.2 can be found in Table 6.1.

6.0.6 Future Research Directions

Future research could explore alternative methods like Unsupervised Learning to cluster different propaganda techniques rather than relying on Supervised Learning and labeled datasets.

A potential research direction could involve utilizing various propaganda datasets to create more propagandistic spans, labeling the techniques with a weak learner, and reapplying them to train additional models, similar to the method employed by Jurkiewicz et al. [JBKG20].

An untested approach includes incorporating Transfer Learning by utilizing labeled datasets within the disinformation segment or incorporating datasets with sentiment knowledge. Using a Cross-Language Model, the Chinese dataset by Chang et al. [CLCL21] could be utilized. The dataset classifies state-sponsored propagandistic Tweets from China with the techniques proposed by Da San Martino et al. [DSMYBC+19]. Moreover,

Table 6.1: Actualized Overall Analysis of the Articles

Publisher	Scraped Articles	Total Sentences	Detected Propaganda	Detected Percentage	Mean Sentences	Propaganda per Article
ABC News	3998	193472	33362	17.24	48.39	8.34
CBS News	772	26346	5469	20.76	34.13	7.08
CNN News	2986	142043	31041	21.85	47.57	10.4
Fox News	4263	157643	34111	21.64	36.98	8.0
Politico	937	47747	10138	21.23	50.96	10.82
Summary	**12956**	**567251**	**114121**	**20.11%**	**43.78**	**8.81**
News Front	1432	30550	6145	20.11	21.33	4.29
Novye Izvestia	1378	47991	9950	20.73	34.83	7.22
Russia Today RU	20216	395496	91005	23.01	19.56	4.5
Sputnik	7784	158793	37717	23.75	20.4	4.85
Tass News	7647	111819	21244	19.0	14.62	2.78
Summary EN	**38457**	**744649**	**166061**	**22.30%**	**19.36**	**4.31**
Russia Today RU	723	34496	9485	27.5%	47.71	13.12
Summary RU	**723**	**34496**	**9485**	**27.50%**	**47.71**	**13.12**

finally, by utilizing the dataset by Dimitrov et al. [DBAS+21], knowledge derived from propagandistic memes could have been used to enhance the model.

Finally, employing ChatGPT to construct a Propaganda Detector using effective prompts rather than programming represents a promising approach that could make Propaganda Detection accessible to stakeholders with limited technical proficiency.

List of Figures

List of Tables

Acronyms

BERT Bidirectional Encoder Representations from Transformers. 18, 19, 28, 32–36, 55, 90, 92

BIO Begin, Inside, Outside. 44

CRF Conditional Random Field. 25, 44

ELMo Embeddings from Language Models. 21, 34, 35

GPT Generative Pre-trained Transformer. 17

GPT-2 Generative Pre-trained Transformer 2. 17, 28

GPT-3 Generative Pre-trained Transformer 3. 19–21, 92

GPT-4 Generative Pre-trained Transformer 4. 20, 92

GPU Graphics Processing Unit. 21, 92

LIWC Linguistic Inquiry and Word Count. 32, 34

LLaMA Large Language Model Meta AI. 21, 92

LLM Large Language Model. 20, 21

LSTM Long Short-Term Memory Network. 15, 28, 34–36

OPT Open Pre-trained Transformer. 20, 55, 90, 92

RoBERTa Robustly Optimized BERT Pre-training Approach. 18, 19, 28–30, 35, 36, 43–45, 47–49, 55, 90, 92

TF-IDF Term Frequency Inverse Document Frequency. 12, 22, 29, 33, 35

XLM Cross Language Model. 19, 28

XLM-R Cross Language Model RoBERTa. 19

Bibliography

[AAO20] Ola Altiti, Malak Abdullah, and Rasha Obiedat. JUST at SemEval-2020 Task 11: Detecting Propaganda Techniques Using BERT Pre-trained Model. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1749–1755, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[AS20] Anastasios Arsenos and Georgios Siolas. NTUAAILS at SemEval-2020 Task 11: Propaganda Detection and Classification with biLSTMs and ELMo. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1495–1501, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[BHL$^+$21] Judit Bayer, Bernd Holznagel, Katarzyna Lubianiec, Adela Pintea, Josephine B. Schmitt, Judit Szakács, and Erik Uszkiewicz. *Disinformation and Propaganda: Impact on the Functioning of the Rule of Law and Democratic Processes in the EU and Its Member States - 2021*. European Parliament, 2021.

[BHR00] Lasse Bergroth, Harri Hakonen, and Timo Raita. A Survey of Longest Common Subsequence Algorithms. In *Proceedings Seventh International Symposium on String Processing and Information Retrieval. SPIRE 2000*, pages 39–48, Los Alamitos, 9 2000. Institute of Electrical and Electronics Engineers.

[BKT20] Verena Blaschke, Maxim Korniyenko, and Sam Tureski. CyberWallE at SemEval-2020 Task 11: An Analysis of Feature Engineering for Ensemble Models for Propaganda Detection. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1469–1480, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[BMR$^+$20] Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language Models Are Few-Shot Learners. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, New York, 2020. Curran Associates Inc.

[Bru01] Ivan Bruha. Pre- and Post-processing in Machine Learning and Data Mining. In Paliouras Georgios, Karkaletsis Vangelis, and Spyropoulos Constantine D, editors, *Lecture Notes in Computer Science*, pages 258–266, Berlin, Heidelberg, 2001. Springer.

[BSA20] Anastasios Bairaktaris, Symeon Symeonidis, and Avi Arampatzis. DUTH at SemEval-2020 Task 11: BERT with Entity Mapping for Propaganda Classification. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1732–1738, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[CCZ06] Olivier Chapelle, Mingmin Chi, and Alexander Zien. A Continuation Method for Semi-Supervised SVMs. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, pages 185–192, New York, 2006. Association for Computing Machinery.

[Cho17] Francois Chollet. *Deep Learning with Python.* Manning Publications, 12 2017.

[CIN20] Anton Chernyavskiy, Dmitry Ilvovsky, and Preslav Nakov. Aschern at SemEval-2020 Task 11: It Takes Three to Tango: RoBERTa, CRF, and Transfer Learning. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1462–1468, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[CKG+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. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online, 7 2020. Association for Computational Linguistics.

[CLCL21] Rong-Ching Chang, Chun-Ming Lai, Kai-Lai Chang, and Chu-Hsing Lin. Dataset of Propaganda Techniques of the State-Sponsored Information Operation of the People's Republic of China, 2021.

[DBAS+21] Dimitar Dimitrov, Bishr Bin Ali, Shaden Shaar, Firoj Alam, Fabrizio Silvestri, Hamed Firooz, Preslav Nakov, and Giovanni Da San Martino. Detecting Propaganda Techniques in Memes. In *Proceedings of the 59th*

Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 6603–6617, Online, 8 2021. Association for Computational Linguistics.

[DCLT19] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, 6 2019. Association for Computational Linguistics.

[DFW20] Guillaume Daval-Frerot and Yannick Weis. WMD at SemEval-2020 Tasks 7 and 11: Assessing Humor and Propaganda Using Unsupervised Data Augmentation. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1865–1874, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[DKS20] Ilya Dimov, Vladislav Korzun, and Ivan Smurov. NoPropaganda at SemEval-2020 Task 11: A Borrowed Approach to Sequence Tagging and Text Classification. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1488–1494, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[DSMBCW$^+$20] Giovanni Da San Martino, Alberto Barrón-Cedeño, Henning Wachsmuth, Rostislav Petrov, and Preslav Nakov. SemEval-2020 task 11: Detection of propaganda techniques in news articles. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1377–1414, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[DSMYBC$^+$19] Giovanni Da San Martino, Seunghak Yu, Alberto Barrón-Cedeño, Rostislav Petrov, and Preslav Nakov. Fine-Grained Analysis of Propaganda in News Articles. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, EMNLP-IJCNLP 2019, pages 5636–5646, Hong Kong, 1 2019. Association for Computational Linguistics.

[DWZ20] Jiaxu Dao, Jin Wang, and Xuejie Zhang. YNU-HPCC at SemEval-2020 Task 11: LSTM Network for Detection of Propaganda Techniques in News Articles. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1509–1515, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[EG20] Vlad Ermurachi and Daniela Gifu. UAIC1860 at SemEval-2020 Task 11: Detection of Propaganda Techniques in News Articles. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1835–1840, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[GBC16] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016.

[GCMK20] Anthony Gillioz, Jacky Casas, Elena Mugellini, and Omar Abou Khaled. Overview of the Transformer-based Models for NLP Tasks. In *15th Conference on Computer Science and Information Systems (FedCSIS)*, pages 179–183, Sofia, 2020.

[GI20] Dmitry Grigorev and Vladimir Ivanov. Inno at SemEval-2020 Task 11: Leveraging Pure Transfomer for Multi-Class Propaganda Detection. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1481–1487, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[HBM$^+$22] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, and others. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022.

[Heb22] Christina Hebel. The Russian Invasion: Putin Settles Accounts with the West. *Der Spiegel*, 2 2022.

[HK00] Arthur E Hoerl and Robert W Kennard. Ridge Regression: Biased Estimation for Nonorthogonal Problems. *Technometrics*, 42(1):80–86, 2000.

[Ho95] Tin Kam Ho. Random decision forests. In *Proceedings of 3rd international conference on document analysis and recognition*, volume 1, pages 278–282, Montreal, 8 1995. Institute of Electrical and Electronics Engineers.

[JBKG20] Dawid Jurkiewicz, Lukasz Borchmann, Izabela Kosmala, and Filip Gralinski. ApplicaAI at SemEval-2020 Task 11: On RoBERTa-CRF, Span CLS and Whether Self-Training Helps Them. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1415–1424, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[JDGD+18] Marcin Junczys-Dowmunt, Roman Grundkiewicz, Tomasz Dwojak, Hieu Hoang, Kenneth Heafield, Tom Neckermann, Frank Seide, Ulrich Germann, Alham Fikri Aji, Nikolay Bogoychev, André F T Martins, and Alexandra Birch. Marian: Fast Neural Machine Translation in C++. In *Proceedings of ACL 2018, System Demonstrations*, pages 116–121, Melbourne, 7 2018. Association for Computational Linguistics.

[JGM20] Yunzhe Jiang, Cristina Garbacea, and Qiaozhu Mei. UMSIForeseer at SemEval-2020 Task 11: Propaganda Detection by Fine-Tuning BERT with Resampling and Ensemble Learning. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1841–1846, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[KB20] Moonsung Kim and Steven Bethard. TTUI at SemEval-2020 Task 11: Propaganda Detection with Transfer Learning and Ensembles. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1829–1834, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[KBG20] Michael Kranzlein, Shabnam Behzad, and Nazli Goharian. Team DoNotDistribute at SemEval-2020 Task 11: Features, Finetuning, and Data Augmentation in Neural Models for Propaganda Detection in News Articles. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1502–1508, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[KGV+14] Subbu Kannan, Vairaprakash Gurusamy, S Vijayarani, J Ilamathi, Ms Nithya, S Kannan, and V Gurusamy. Preprocessing techniques for text mining. In *International Journal of Computer Science and Communication Networks*, volume 5, pages 7–16, 9 2014.

[KGY20] Gangeshwar Krishnamurthy, Raj Kumar Gupta, and Yinping Yang. SocCogCom at SemEval-2020 Task 11: Characterizing and Detecting Propaganda Using Sentence-Level Emotional Salience Features. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1793–1801, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[KMS+19] Johannes Kiesel, Maria Mestre, Rishabh Shukla, Emmanuel Vincent, Payam Adineh, David Corney, Benno Stein, and Martin Potthast. SemEval-2019 Task 4: Hyperpartisan News Detection. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 829–839, Minneapolis, 6 2019. Association for Computational Linguistics.

[KTP20] Anders Kaas, Viktor Torp Thomsen, and Barbara Plank. Team DiSaster at SemEval-2020 Task 11: Combining BERT and Hand-crafted Features for Identifying Propaganda Techniques in News. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1817–1822, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[LBH15] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 5 2015.

[LCG^{+}19] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. *arXiv preprint arXiv:1909.11942*, 2019.

[LMP01] John D Lafferty, Andrew McCallum, and Fernando C N Pereira. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pages 282–289, San Francisco, 2001. Morgan Kaufmann Publishers Inc.

[LWLQ22] Tianyang Lin, Yuxin Wang, Xiangyang Liu, and Xipeng Qiu. A survey of transformers. *AI Open*, 3:111–132, 2022.

[LX20] Jinfen Li and Lu Xiao. syrapropa at SemEval-2020 Task 11: BERT-based Models Design for Propagandistic Technique and Span Detection. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1808–1816, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[MCCD13] Tomas Mikolov, Kai Chen, Gregory S Corrado, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space. In *1st International Conference on Learning Representations*, Scottsdale, 5 2013.

[MCG16] Michael McTear, Zoraida Callejas, and David Griol. Spoken Language Understanding. In *The Conversational Interface: Talking to Smart Devices*, pages 161–185. Springer, Cham, 2016.

[MG13] Neha Mehra and Surendra Gupta. Survey on multiclass classification methods. *International Journal of Computer Science and Information Technologies*, 4(4):572–576, 2013.

[Mil23] Ron Miller. Sam Altman: Size of LLMs won't matter as much moving forward, 4 2023.

[MMOM20] Gaku Morio, Terufumi Morishita, Hiroaki Ozaki, and Toshinori Miyoshi. Hitachi at SemEval-2020 Task 11: An Empirical Study of Pre-Trained Transformer Family for Propaganda Detection. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1739–1748, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[MPS20] Matej Martinkovic, Samuel Pecar, and Marian Simko. NLFIIT at SemEval-2020 Task 11: Neural Network Architectures for Detection of Propaganda Techniques in News Articles. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1771–1778, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[NFR+22] Nic Newman, Richard Fletcher, Craig T Robertson, Kirsten Eddy, and Rasmus Kleis Nielsen. Reuters Institute digital news report 2022. 2022.

[NS16] Krystyna Napierala and Jerzy Stefanowski. Types of minority class examples and their influence on learning classifiers from imbalanced data. *Journal of Intelligent Information Systems*, 46(3):563–597, 2016.

[Ope23] OpenAI. GPT-4 Technical Report. *ArXiv*, abs/2303.08774, 2023.

[PA01] Anthony R Pratkanis and Elliot Aronson. *Age of Propaganda: The Everyday Use and Abuse of Persuasion*. Henry Holt & Co, 2001.

[PBJB15] James W Pennebaker, Ryan L Boyd, Kayla Jordan, and Kate Blackburn. The development and psychometric properties of LIWC2015. Technical report, 2015.

[PCD20] Andrei Paraschiv, Dumitru-Clementin Cercel, and Mihai Dascalu. UPB at SemEval-2020 Task 11: Propaganda Detection with Domain-Specific Trained BERT. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1853–1857, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[PGM+19] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In H Wallach, H Larochelle, A Beygelzimer, F d Alché-Buc, E Fox, and R Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[PNI^{+}18] Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep Contextualized Word Representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, 6 2018. Association for Computational Linguistics.

[Pow11] David M W Powers. Evaluation: From precision, recall and f-measure to roc., informedness, markedness & correlation. *Journal of Machine Learning Technologies*, 2(1):37–63, 2011.

[PP20] Maia Petee and Alexis Palmer. UNTLing at SemEval-2020 Task 11: Detection of Propaganda Techniques in English News Articles. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1847–1852, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[PSA20] Rajaswa Patil, Somesh Singh, and Swati Agarwal. BPGC at SemEval-2020 Task 11: Propaganda Detection in News Articles with Multi-Granularity Knowledge Sharing and Linguistic Features Based Ensemble Learning. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1722–1731, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[PSM14] Jeffrey Pennington, Richard Socher, and Christopher D Manning. GloVe: Global Vectors for Word Representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014.

[RJR^{+}20] Mayank Raj, Ajay Jaiswal, Rohit R.R, Ankita Gupta, Sudeep Kumar Sahoo, Vertika Srivastava, and Yeon Hyang Kim. Solomon at SemEval-2020 Task 11: Ensemble Architecture for Fine-Tuned Propaganda Detection in News Articles. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1802–1807, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[RNSS18] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. 2018.

[RS04] Jennifer Rowley and Frances Slack. Conducting a literature review. *Management Research News*, 27(6):31–39, 1 2004.

[RWC^{+}19] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[SDCW19] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *Computing Research Repository*, 2019.

[SF11] Carlos N Silla and Alex A Freitas. A survey of hierarchical classification across different application domains. *Data Mining and Knowledge Discovery*, 22(1):31–72, 2011.

[SP97] Mike Schuster and Kuldip K. Paliwal. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11), 1997.

[SSKM20] Paramansh Singh, Siraj Sandhu, Subham Kumar, and Ashutosh Modi. newsSweeper at SemEval-2020 Task 11: Context-Aware Rich Feature Representations for Propaganda Classification. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1764–1770, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[TLI+23] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. LLaMA: Open and Efficient Foundation Language Models, 2 2023.

[vEH20] Jesper E van Engelen and Holger H Hoos. A survey on semi-supervised learning. *Machine Learning*, 109(2):373–440, 2020.

[VMC20] Ekansh Verma, Vinodh Motupalli, and Souradip Chakraborty. Transformers at SemEval-2020 Task 11: Propaganda Fragment Detection Using Diversified BERT Architectures Based Ensemble Learning. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1823–1828, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[VSP+17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[WH00] Rüdiger Wirth and Jochen Hipp. CRISP-DM: Towards a Standard Process Model for Data Mining. *Proceedings of the 4th International Conference on the Practical Applications of Knowledge Discovery and Data Mining*, 2000.

[WKW16] Karl Weiss, Taghi M Khoshgoftaar, and DingDing Wang. A survey of transfer learning. *Journal of Big Data*, 3(1):9, 2016.

[Wol92] David H Wolpert. Stacked generalization. *Neural Networks*, 5(2):241–259, 1992.

[XDH+20] Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. Unsupervised data augmentation for consistency training. *Advances in Neural Information Processing Systems*, 33:6256–6268, 2020.

[YDY+19] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V Le. XLNet: Generalized Autoregressive Pre-training for Language Understanding. In *Advances in neural information processing systems*, volume 32, Red Hook, 2019. Curran Associates Inc.

[YtLYc05] Zhang Yun-tao, Gong Ling, and Wang Yong-cheng. An improved TF-IDF approach for text classification. *Journal of Zhejiang University-SCIENCE A*, 6(1):49–55, 2005.

[Zhu08] Xiaojin Zhu. Semi-Supervised Learning Literature Survey. *University of Wisconsin-Madison Department of Computer Sciences*, 12 2008.

[ZRG+22] Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. OPT: Open Pre-trained Transformer Language Models. *arXiv preprint arXiv:2205.01068*, 2022.

[ZWT02] Zhi-Hua Zhou, Jianxin Wu, and Wei Tang. Ensembling neural networks: Many could be better than all. *Artificial Intelligence*, 137(1):239–263, 2002.

[ZWYJ21] Liu Zhuang, Lin Wayne, Shi Ya, and Zhao Jun. A Robustly Optimized BERT Pre-training Approach with Post-training. In *Proceedings of the 20th Chinese National Conference on Computational Linguistics*, pages 1218–1227, Huhhot, 8 2021. Chinese Information Processing Society of China.

Erklärung zur Verfassung der Arbeit

Vitalij Hein, B.Sc.

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Wien, 6. August 2023

V. Hein

Vitalij Hein

Acknowledgements

After spending a year immersed in my thesis work, I want to acknowledge those who made this journey possible.

Ao.Univ.Prof. Mag. Dr. Dieter Merkl, my supervisor, has been an incredible guiding force throughout this time. He patiently helped me navigate through the complexities of scientific work, always knowing when to apply a gentle push to keep me motivated while respecting my personal work rhythm. I would also be remiss not to mention his invaluable travel advice for Italy.

My heartfelt thanks goes to Vanessa Raffler as well. Her assistance, filled with valuable ideas and improvements, was crucial while developing the Propaganda Detection Model. I am particularly grateful for her patient ear when I was burdened with concerns. Her presence made the process significantly less daunting.

Lastly, I owe an immeasurable debt of gratitude to my parents. Their unwavering emphasis on the importance of education has always inspired us to strive for more in life. Their support instilled a drive within us to reach for our ambitions, and for that, I am eternally grateful.

Abstract

During different decades, propaganda was a vital technique to manipulate human opinions subtly. With the rise of mass media, subtle opinions can be incepted by various sources without being detected by the reader. A whole population can be manipulated into believing certain aspects this way. This thesis investigates the issue of propaganda in news articles. By applying the CRISP-DM research methodology, the study analyzes propaganda techniques in Russian and American news contexts, focusing on media coverage of the attack on the Ukrainian. The research presents a novel Propaganda Detection Model that utilizes Ensemble Learning. The model combines multiple Transformer Models, including BERT, OPT, and RoBERTa, to detect 18 propaganda techniques. A key finding is the effectiveness of Meta Classification with Model Stacking, surpassing Ensemble Averaging and other methods, achieving a Micro F1-Score of 0.78, indicating a significant level of accuracy in Propaganda Detection. The research reveals the potential of the Propaganda Detection Model to counteract propaganda in news articles, contributing to preserving democratic processes. It contributes to understanding the utilization of propaganda in Russian and American news contexts by analyzing its monthly usage from January 2022 to April 2023 and comparing the distribution of propaganda techniques.

Kurzfassung

In der Vergangenheit war Propaganda eine entscheidende Technik, um die Meinungen von Menschen subtil zu manipulieren. Mit dem Aufstieg der Massenmedien können subtile Meinungen von verschiedenen Quellen vermittelt werden, ohne dass der Leser dies erkennt. Auf diese Weise kann eine ganze Bevölkerung dazu manipuliert werden, bestimmte Aspekte zu glauben. Diese Thesis untersucht das Thema Propaganda in Nachrichtenartikeln. Durch Anwendung der CRISP-DM-Forschungsmethodik analysiert die Studie Propaganda-Techniken im russischen und amerikanischen Nachrichten, mit Fokus auf die Medienberichterstattung über den Angriff auf die Ukraine. Die Forschung präsentiert ein Propaganda Detection Model, das Ensemble Learning verwendet. Das Modell kombiniert mehrere Transformer Models, einschließlich BERT, OPT und RoBERTa, um 18 Propaganda-Techniken zu erkennen. Eine zentrale Erkenntnis ist die Wirksamkeit von Meta Classification mit Model Stacking, was Ensemble Averaging und andere Methoden übertrifft und einen Micro F1-Score von 0,78 erreicht, was auf ein hohes Maß an Genauigkeit bei der Propaganda-Erkennung hindeutet. Die Forschung zeigt das Potenzial des Propaganda Detection Model, Propaganda in Nachrichtenartikeln entgegenzuwirken und damit demokratische Prozesse zu bewahren. Sie trägt zum Verständnis der Nutzung von Propaganda im russischen und amerikanischen Nachrichtenkontext bei, indem sie deren monatliche Nutzung von Januar 2022 bis April 2023 analysiert und die Verteilung der Propaganda-Techniken vergleicht.

Contents

CHAPTER 1

Introduction

1.1 Motivation & Problem Statement

With the rise of digitalization, news consumption changed totally. The Digital News Report [NFR+22] states that traditional media consumption, like TV and print, declined further in 2021, but online and social consumption do not fill the gap. News consumption is declining, and the overall interest in news is also falling across the investigated markets. More precisely, the interest has fallen from 63% in 2017 to 51% in 2022 [NFR+22].

According to an EU study about disinformation and propaganda, updated in 2021, recent developments in disinformation campaigns nowadays do not care about the true/false dichotomy. The goal of modern disinformation campaigns is not to convince their target audience of alternative narratives, but instead, they aim to create or deepen social division. Many campaigns aim to pit social groups against each other by distributing polarised content tailored to both sides. Also, distrust is created by putting forward so-called alternative facts. The growing distrust towards media, news, and politics can have far-reaching consequences for democracy, like reduced civic participation in the political process or even decreased acceptance of democratic legitimacy in the eyes of citizens [BHL+21].

Modern propaganda works with the intelligent use of images and symbols to appeal to human biases and emotions. Propaganda is a communication technique that makes a recipient believe a foreign opinion, delimiting from persuasion: The recipient is manipulated to believe an opinion that would be unacceptable under other circumstances [PA01].

With the Russian attack on Ukraine, Russian propaganda became a broadly discussed topic in Germany. Its interest can be shown with the help of Google Trends[1]. Google

[1] https://trends.google.com (last accessed: 29 July 2022)

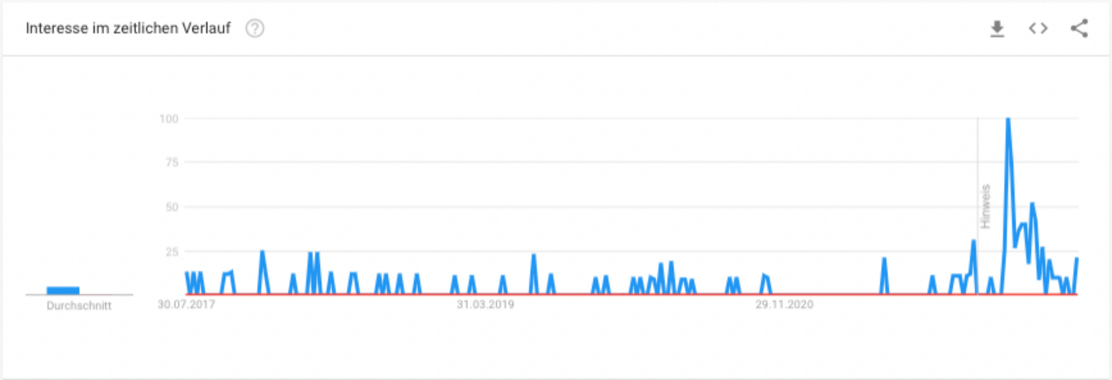


Figure 1.1: Interest in American and Russian propaganda in Germany from 2017 showing the consequent 5 years. Data source: Google Trends

Trends aggregates all search requests that their users do. All the distinct search requests are then split into coherent topics. By checking Google Trends, interest in topics can be evaluated. Interest, in this case, is the aggregated number of search requests for a specific topic. Interest is indicated in a range between 0-100 for any given date. When checking Google Trends with the search strings "Russia Propaganda" and "America Propaganda", "5 years from now" and only in "Germany" reveals an interesting fact: As shown in Fig. 1.1, the interest in American propaganda is indicated as non-existent by Google Trends, without any further data displayed. In contrast, interest in Russian propaganda had a massive spike around February 2022 and slowly declined to lower relevancy. The blue line represents an interest in Russian Propaganda, while the red line represents American Propaganda.

Based on this observation, propaganda is a phenomenon that is important in the context of Russia but not in the context of the USA. However, does this mean that states like the USA do not use propaganda techniques? Since Google Trends only analyzes search requests, it only shows the collective focus and reveals potential unconscious bias in the thinking processes of a group or society.

1.2 Research Questions

The following results are expected to be answered when researching the utilization of propaganda techniques in Russian and American news.

1.2.1 In which way can the findings of the SemEval 2020 Task 11 [DSMBCW+20] be combined, and to which degree in terms of F1-Score will this combination be able to compete with the results of the Technique Classification subtask?

Since about 25 teams published a paper, many different approaches were tested, and therefore combining different ideas and approaches could lead to an even better model. By examining those approaches, a Propaganda Detection Model is created and gradually enhanced with strategies from the other teams.

1.2.2 What are the similarities and differences in using the different propaganda techniques when looking at Russian and American news articles?

After creating a Propaganda Detection Model, the model will be used against an unlabelled, mined dataset of articles from Russian and American news sources. The model should be able to classify the sentences of the different news articles and show if a propaganda technique is used there. It is possible to show which techniques both countries use and to what extent. For example, a possible outcome could be that American articles utilize *Whataboutism* more extensively than Russian articles. In contrast, Russians could use *Reductio ad Hitlerum* more often since they claim to denazify Ukraine [Heb22]. Such understanding can lead to a better understanding of disinformation and the usage of propaganda, which is an essential suggestion in European research on disinformation and propaganda [BHL+21].

1.2.3 How did the usage of different propaganda techniques change during the timeline of the Ukrainian war?

There are 18 different propaganda techniques the model can classify. The scraped data from Russian and American news outlets contains a publishing date for every article. This publishing date helps to sort the classified propaganda techniques based on their release date. It is expected to see a different distribution of the techniques during the war proceeding.

1.3 Methodology and Approach

The CRISP-DM methodology is used to build the Propaganda Detection Model.

CRISP-DM, short for Cross Industry Standard Process for Data Mining, addresses Data Mining as a creative process that requires several different skills and knowledge. Before CRISP-DM, there was no standard framework for Data Mining projects. The methodology defines an explicit process model that provides a framework for such projects. It is described in terms of the hierarchical process model, comprising four abstraction levels, ranging from general to specific. It is built upon phases, generic tasks, specialized

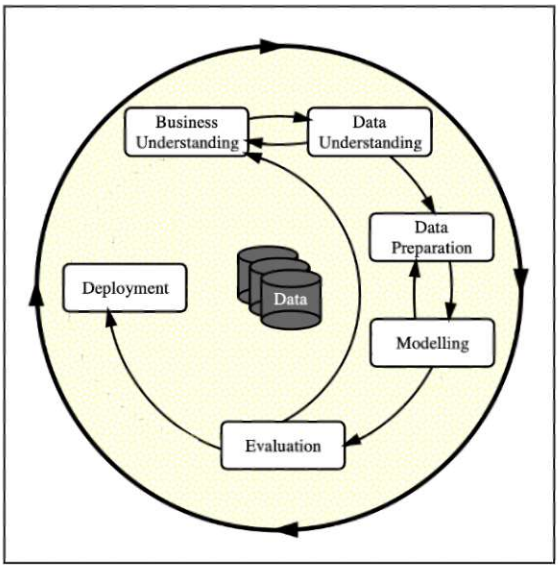


Figure 1.2: The CRISP-DM Lifecycle [WH00]

tasks, and process instances. The CRISP-DM Reference Model gives an overview of the essential phases, their tasks, and their expected output [WH00]. While the whole CRISP-DM process is extensive, some parts are skipped during the practical part since they do not apply to the thesis.

1.3.1 Business Understanding

The first phase in CRISP-DM is Business Understanding, which aims to gather a first understanding of the problem. Since a business does not instruct this thesis, the disinformation and propaganda study conducted by the European Parliament [BHL+21] is used to gather Business Understanding.

1.3.2 Data Understanding

The second phase of CRISP-DM is dedicated to Data Understanding. Data is split into the Propaganda Technique Corpus [DSMYBC+19], and the Analysis Dataset. This phase hosts statistical analysis of the datasets.

1.3.3 Data Preparation

During Data Preparation, the datasets are prepared for the consequent modeling phase, and different features are generated.

1.3.4 Modeling

The Modeling phase is all about building the Propaganda Detection Model. First, the Modeling Technique is selected, the Test Design generated, the Model Built, and the Quality Assessed.

1.3.5 Evaluation

The evaluation will occur on the Micro F1-Score since the Propaganda Technique Corpus has an uneven class distribution and a balance between Precision and Recall is needed. Also, the SemEval challenge used the Micro F1-Score as their prioritized measure for performance [DSMBCW$^+$20].

1.3.6 Deployment

The last phase in CRISP-DM is the Deployment phase, where the Deployment, Monitoring, and Maintenance are planned. The model will be made public by deploying it to Github[2]. This way, further research is encouraged.

1.4 Structure of Thesis

The first chapter briefly introduces Propaganda Detection's importance, the Research Questions, and the chosen Research Method. Chapter 2 explores the Theoretical Backgrounds of Deep Learning with a focus on Transformer-based Language Models, explaining the key concepts and theories that serve as the foundation for Literature Analysis in Chapter 3, where the SemEval 2020 Task 11 findings are summarized and analyzed. Chapter 4 then documents the execution of the CRISP-DM methodology, while in Chapter 5, the results of the three Research Questions are disclosed. The last chapter hosts a Discussion on the retrieved results of the Propaganda Detection Model, revealing Limitations and Future Research directions and an additional experiment on Russian-language articles.

[2] https://github.com (last accessed: 08 August 2022)

CHAPTER 2

Theoretical Backgrounds

This chapter serves as an overview of important theoretical concepts covered in this work.

2.1 SemEval 2020 Task 11: Propaganda Detection

The SemEval 2020 Task 11 offers a fine-grained analysis of propaganda in news articles. The challenge was split into two parts, Span Identification and Technique Classification. Further, the challenge was divided into two phases. Only training and development data were available during the first phase, while no propaganda labels were provided for the development data. The participants tried to achieve the best performance on the development data. The participants could make unlimited submissions on the development dataset, seeing the impact of their modifications and their performance compared to other participants. Gold labels were released during the second phase for the development data, and a third test dataset was released. This time the participants did not get any feedback on their performance on the test data. After the challenge was finished, only the submission system of the first phase was left open [DSMBCW$^+$20].

2.1.1 Span Identification and Technique Classification Tasks

Since the initial goal of the challenge was first to identify propagandistic spans in a sentence and second to classify them into a given set of labels, the challenge was organized into two subtasks. In the Span Identification subtask, participants were given a plain-text document and had to identify the fragments containing propaganda. In the Technique Classification subtask, text snippets identified as propaganda were already given. The participants had to build a Multi-class Classification model to predict the propaganda technique of the given text span [DSMBCW$^+$20].

2.1.2 Provided Dataset

The hosts created a labeled dataset with 18 distinct propaganda techniques annotated. The training and development datasets include 446 articles and over 400.000 tokens from 48 news outlets. The classes *Whataboutism*, *Straw Man*, and *Red Herring* were merged into one class since each is underrepresented. The three techniques share similarities since each tries to confuse a reader by bringing Attention to something irrelevant, away from the actual problem. Also, *Bandwagon* and *Reductio ad Hitlerum* were merged, both trying to approve or disapprove of an action or idea by pointing to something unpopular or popular [DSMBCW$^+$20].

2.1.3 Propaganda Techniques

For a quick overview, the following part summarizes each propaganda technique defined by Da San Martino et al. [DSMBCW$^+$20].

1. **Loaded Language:** Using strong emotional words to influence an audience.
2. **Name Calling or Labeling:** Labeling a subject of interest as something the audience hates or loves to connect the emotion to the subject of interest.
3. **Repetition:** Repeating a message multiple times until the message sticks with the target audience.
4. **Exaggeration or Minimization:** Trying to make things larger or smaller by repeating them excessively.
5. **Doubt:** Bringing someone's or something's trustworthiness into question.
6. **Appeal to Fear or Prejudice:** Attempting to gain support for an idea by stirring up fear and panic among the population about an alternative, possibly based on prejudices.
7. **Flag-waving:** Exploiting strong national sentiments or relating to a population group, like race, gender, or political preference, to justify or promote an initiative or concept.
8. **Causal Oversimplification:** Assuming a single cause when several causes are related to a problem. It also includes scapegoating: blaming one person or a group without considering the complexity of a problem.
9. **Slogans:** A short statement that can contain labeling and stereotyping.
10. **Appeal to Authority:** Assertion that a claim is true just because a valid authority or expert supports it, without further evidence.
11. **Black-and-white Fallacy or Dictatorship:** Pretending that two choices are the only options when there are more.

12. **Thought-terminating Cliche:** Words or phrases that prevent critical thought and meaningful debate on a particular topic.

13. **Whataboutism:** Discredit the other party's position by accusing them of hypocrisy while not directly disproving their arguments.

14. **Reductio ad Hitlerum:** Convincing an audience to dislike an action or idea by implying the idea is popular with groups that the target audience despises.

15. **Red Herring:** Bringing irrelevant material into the discussion such that Attention is distracted from the actual issues.

16. **Bandwagon:** Trying to convince the audience to participate and follow the course because "everyone else is doing the same thing".

17. **Obfuscation, Intentional Vagueness, Confusion:** Intentionally using vague language to let the audience make their interpretation.

18. **Straw Man:** When the opponent's claim is substituted by a related one but then refuted in place of the initial one.

Given the Propaganda Technique Corpus with the different propaganda techniques, the dataset serves as the foundation of the Propaganda Detection Model and the subsequent analysis.

2.2 What is Deep Learning?

Deep Learning, a significant subset of Machine Learning, has revolutionized various fields by enabling machines to process images, music, speech, audio, and sequential data like text. At its core, Deep Learning refers to the specific architecture of a Neural Network-based Learning System, which utilizes computational models with multiple Processing Layers to process raw input data efficiently. These layers handle different aspects of the input data and contribute to the system's overall performance [LBH15].

Before the emergence of Deep Learning, conventional Machine Learning techniques relied heavily on expertly designed Feature Extractors, which required extensive domain knowledge of a dataset paired with expertise in Machine Learning. This process involved crafting a Feature Vector, which was then input into a simple classifier to detect patterns in the data. The engineering of this Feature Vector was a labor-intensive task, eventually streamlined by Representation Learning. Representation Learning is a set of methods allowing a machine to automatically discover the necessary representations for Classification or detection tasks. Consequently, the previously hand-crafted Feature Vector is generated automatically, eliminating the need for deep domain expertise in the input data. This ability to learn representations is crucial in differentiating Deep Learning from conventional Machine Learning techniques [LBH15].

The Multilayer Architecture of Deep Learning systems is one of its distinguishing features. In a Deep Learning environment, Learning occurs chained, where raw input data is fed into the first layer, creating a more abstract representation of the input by applying simple yet non-linear methods. The representations become increasingly abstract as the data passes through each layer, enabling the machine to use highly complex functions to extract knowledge from the data. Upon closer examination of the layers, their complexity is manageable: They comprise simple modules capable of solving complex problems when interconnected [LBH15].

During Deep Learning, the goal is to generate representations of data with multiple levels of abstraction. These representations can be achieved by discovering intricate structures in large data sets using the Backpropagation algorithm. This algorithm indicates how the Learning System should change its internal parameters to optimize the representations in the different Processing Layers. It is important to note that, in this process, each layer learns from the previous layer, so Learning occurs chained. At its core, the Backpropagation procedure is the practical application of the chain rule for derivatives. This rule states that the gradient of an objective concerning the module input can be computed by working backward from the gradient concerning the module output. The Backpropagation equation can be repeatedly applied to propagate gradients through all modules, starting from the prediction output and returning to the module where the raw input data is introduced [LBH15].

2.3 Various Concepts in Deep Learning

2.3.1 Preprocessing

Deep Learning models need a specific representation of input data to consume and learn from the data. Whenever the original format of the raw data is not usable by a model, Preprocessing techniques are applied [GBC16].

Stopwords

Stopwords are common words, such as *and*, *are*, and *this*, that appear frequently in documents but do not contribute to the sentence context. Their high frequency of occurrence can hinder a model from understanding the content. Thus those words can be removed during Preprocessing [KGV$^+$14].

Stemming

Stemming is a text processing technique used in Information Retrieval, which reduces variant forms of a word to their standard stem. For example, *presentation*, *presented*, and *presenting* are stemmed to *present*. It operates on the assumption that searching for one form of a word implies an interest in all its variants [KGV$^+$14].

Tokenization

Tokenization converts a text block, such as words or phrases, into smaller units called tokens. The process helps identify meaningful keywords, ensuring document consistency by addressing punctuation marks, different number and time formats, and standardizing abbreviations and acronyms [KGV+14].

Embedding Generation

When working with Transformer-based Language Models, it is crucial to perform Tokenization to generate a list of subwords. Each subword is then mapped to a unique integer from the model's vocabulary. In the next step, the integer vector is converted into a vector representation used as an input for the Transformer model [VSP+17]. The mapping is different with every Transformer-based Language Model since every model has its unique vocabulary.

2.3.2 Hyperparameter Optimization

Building a Deep Learning model requires different architectural decisions, like optimizing Hyperparameters, the number of layers, filters per layer, and the type of Activation Function. Initial choices may only sometimes be optimal due to the absence of formal rules [Cho17].

Automatic Hyperparameter Optimization has been developed to efficiently explore possible decisions, reducing the need for time-consuming manual adjustments. Optimization involves automatically selecting Hyperparameters, building and training the model, evaluating performance on validation data, and then iterating this process with a new set of Hyperparameters. The new set of Hyperparameters is determined based on past validation performance. The selection is crucial and can be done by employing various techniques, like Bayesian Optimization or Random Search [Cho17].

Contrary to training model weights, updating Hyperparameters requires creating and training a new model, as Hyperparameters are typically discrete and non-differentiable. Random Search is commonly used for Hyperparameters Optimization, but different Python libraries provide more efficient solutions [Cho17].

However, a significant concern is Overfitting to the validation dataset as Hyperparameters are updated based on validation data. Ultimately, Hyperparameter Optimization is essential for achieving top-performing models or succeeding in Machine Learning competitions [Cho17].

2.3.3 Data Augmentation

Deep Learning requires a large amount of labeled data to work well. During Supervised Data Augmentation, a sentence is modified to resemble the original example. This way, the augmented example can still be labeled under the same category as the original sentence. The synthetic data is then used with the original data to train a Classification model [XDH$^+$20].

Xie et al. [XDH$^+$20] proposed a different method, termed Unsupervised Data Augmentation. The procedure can be summarized by first computing an output distribution from an input sentence. In parallel, a noised version is created by injecting noise, and the divergence between the two versions is minimized to calculate a final loss. This way, the model gets insensitive to changes in the input space. Simultaneously minimizing the consistency loss gradually propagates label information from labeled examples to unlabeled ones.

Another possible Advanced Data Augmentation strategy for Text Classification is Backtranslation. A sentence is translated into a target language and then returned to the original language. This way, some words or even the sentence structure may change, but the semantics of the original sentence are preserved. Another possible strategy for Text Classification is word replacing with Term Frequency Inverse Document Frequency (TF-IDF). This strategy is helpful if specific keywords with high importance must be preserved in the augmented variation. Words with lower importance, thus with low TF-IDF scores, can be replaced with other low-score words [XDH$^+$20]. Another possible approach is Synonym Replacement [DFW20].

2.3.4 Inference

Inference refers to the process where relationships between variables are deduced. Unknown or hidden variables can be predicted by examining known values or observed variables. This process is often vital for performing other tasks or implementing learning rules. These learning rules are typically founded on the Principle of Maximum Likelihood, which aids in determining the most probable outcomes. Inference, therefore, involves predicting values or probabilities of unknown variables, given the known values of other variables [GBC16].

2.3.5 Postprocessing

After a Deep Learning model has produced its predictions, these results can be adjusted during Postprocessing. This stage can involve pruning, filtering, or integrating additional knowledge [Bru01].

2.3.6 Generalization, Over- and Underfitting

Generalization describes performing well on new, unseen inputs, not just those used during training. The training dataset is used to compute and minimize the training error. Generalization also aims to minimize the validation error, the expected error on a new input [GBC16].

The error is typically estimated by measuring the model's performance on a separate validation dataset. In the Machine Learning process, parameters are not predetermined.

Instead, the training dataset is used to adjust the parameters to lower the training error, and then the validation dataset is evaluated. As a result, the expected validation error is usually equal to or higher than the training error [GBC16].

A Machine Learning algorithm's success is determined by its ability to reduce the training error and the gap between the training and validation error. These are directly related to the key challenges in Machine Learning: Underfitting, where the model cannot achieve a low enough training error, and Overfitting, where the gap between training and validation error is too high. These challenges represent the factors that Machine Learning models must balance to be effective [GBC16].

2.3.7 Cross-Validation

The network evaluation can be achieved by adjusting various parameters, such as the number of epochs used for training, by dividing data into training and validation datasets. When dealing with a smaller dataset, the validation dataset could be limited, perhaps around 100 examples, leading to a potentially high variance in validation scores. This significant fluctuation in validation scores, dependent on which data points are selected for validation and training, can hinder a reliable model evaluation. In such instances, it is recommended to employ K-fold Cross-Validation. This method involves dividing the data into K partitions creating K identical models, and training each one on K–1 partitions while evaluating the remaining partition. The model's validation score is then determined as the average of the K different validation scores obtained, offering a comprehensive evaluation of the model's performance across different data partitions [Cho17].

2.3.8 Ensembling

Ensembling is a technique that involves training multiple different models on the same data and combining their predictions to make a final prediction. The main goal of Ensembling is to produce a final model that is more accurate and robust than any of the individual models [ZWT02].

2.3.9 Meta Classification - Stacked Generalization

Stacked Generalization is a form of Meta Classification. It is introduced as a methodology to reduce the Generalization Error of one or multiple models. The concept is based on determining the bias of the models relative to a given learning set. This discernment is achieved through a second round of Generalization, which operates in a distinct space. In this secondary space, the inputs are typically the predictions of the original models, which have been trained on a portion of the learning dataset and attempt to predict the remainder. The output of the second space is typically the correct prediction [Wol92].

2.4 Deep Learning Architectures

2.4.1 Feedforward Neural Network

A typical architecture in Deep Learning is a Feedforward Neural Network. These networks learn to map a fixed-size input to a fixed-size output, for example, a probability (output) for a category (input). To traverse the layer structure, a set of units compute a weighted sum of their inputs from the previous layer and pass the result through a non-linear function. The units between the Input and Output Layers are called Hidden Layers. They distort the input data non-linearly so that the different categories become linearly separable by the Output Layer [LBH15]. In Deep Learning, non-linear functions are often called Activation Functions.

Different kinds of Output Layers exist, like the Linear Layer or the Softmax Layer. A Linear Layer is a relatively simple example of an Output Layer, and it linearly splits the distribution into two distinct parts. The layer can only learn Linear Relations, making them useless for learning non-linearity. The Linear Layer reduces the previous layers' dimensions to ease data interpretation during Learning. It takes a flattened one-dimensional vector as input and is multiplied by a weight matrix, yielding the output feature [GBC16].

A Softmax Layer is used whenever a Probability Distribution is needed. Most often, Softmax Layers is used as the output of a classifier. However, they can also be used inside the model itself if the model has to choose between multiple options between some internal variables. When applying Softmax for Multi-class Classification, each output probability for the distinct classes has to lie between 0 and 1. The different output probabilities are outputted as a vector, with the specialty that all probabilities sum up to 1. The Softmax Function outputs zero for a class if it clearly cannot be classified as such, while one corresponds to absolute certainty [GBC16]. Simple Feedforward Neural Networks are often also called Multilayer Perceptrons.

2.4.2 Convolutional Neural Networks

A Convolutional Neural Network is a specific type of Feedforward Network that can generalize much better than networks with full connectivity between adjacent layers. Convolutional Neural Networks process data structured in multiple arrays. The four critical concepts behind Convolutional Neural Networks include Local Connections, Shared Weights, Pooling, and the Inclusion of Multiple Layers. The typical architecture consists of a series of stages, with the first stages typically being Convolutional and Pooling Layers. The Convolutional Layer has units organized in Feature Maps, and each unit connects to its previous layer through local patches within the Feature Map. This connection is established through a set of weights. While the primary role of the Convolutional Layer is to detect local conjunctions of features from the previous layer, the Pooling Layer merges semantically similar features. These merges can be done by computing the maximum of a local patch of units in one or a few Feature Maps. Another alternative is Neighboring Pooling, where the input is shifted by more than one row or column. Convolution Stages typically consist of two to three components: Stacked Non-Linearity and Pooling and Convolutional and Fully-connected Layers. Finally, Backpropagation optimizes the parameters for training. Convolutional Neural Networks are particularly well-suited for Image Recognition and processing tasks due to their ability to capture spatial information and hierarchical patterns in data [LBH15].

2.4.3 Recurrent Neural Networks

Recurrent Neural Networks were one of the main architectures that benefited from the introduction of Backpropagation. This type of network is used for tasks that involve sequential inputs, such as speech and language. Recurrent Neural Networks process only one input sequence at a time. In their Hidden Layers, Recurrent Neural Networks contain a State Vector, which preserves information about past elements of the sequence. This recurrent connection enables the network to maintain a memory of previous inputs, allowing it to learn and model temporal dependencies in the data. All layers share the same weights, supporting their primary purpose in learning long-term dependencies. However, it is difficult for Recurrent Neural Networks to store information for an extended period. This limitation can be addressed by incorporating explicit memory components [LBH15].

2.4.4 Long Short-Term Memory Network

One approach to incorporating explicit memory is the Long Short-Term Memory Network (LSTM). The LSTM architecture uses specialized Hidden Layers, which enable a Recurrent Neural Network to remember inputs for a long time. A memory cell acts like an accumulator, connecting the current input sequence to the subsequent one. The memory cell maintains its real-valued state and accumulates external signals. The Self-Connection is gated by another unit that learns when to clear the memory content. The LSTM is an improved version of the Recurrent Neural Network, which tackles the problems of exploding and vanishing gradients during Backpropagation [LBH15]. A drawback of LSTMs is their lacking knowledge of future elements of the input sequence. Bidirectional LSTMs was introduced by Schuster and Paliwal [SP97] to address this problem. These networks consist of two separate LSTMs, one processing the input sequence in the forward direction, the other in the backward direction. The Hidden States of both LSTMs are combined at each computation step to make predictions. This way, both past and future context is captured.

2.5 Transformers in Deep Learning

2.5.1 The first Transformer Model

Introduced by Vaswani et al. [VSP$^+$17] in 2017, the Transformer architecture revolutionized Sequence Transduction models. Before this, prevailing models relied on intricate

Recurrent or Convolutional Neural Networks combined with an Encoder and Decoder. As the most effective models featured an Attention Mechanism linking the Encoder and Decoder, the authors presented a novel network architecture centered exclusively on the Attention Mechanism, making complex Recurrent and Convolutional Networks obsolete. By minimizing sequential computation, Transformers can better comprehend dependencies between distant positions, albeit at the expense of reduced effective resolution. This issue is mitigated through Multi-Head Attention.

The original Transformer model employs the Encoder-Decoder Structure, integrating Stacked Self-Attention and a Fully-connected Layer for both Encoders and Decoders. The Encoder within the Transformer architecture converts an input sequence of symbol representations into a series of continuous representations. Subsequently, the Decoder processes the continuous representation one element at a time, generating an output sequence [VSP$^+$17]. Each representation is created incrementally, reflecting the autoregressive property of Transformer models. Every new output sequence utilizes the latent representation of the previously formed sequence during generation [GCMK20]

2.5.2 Attention in Transformers

The Attention function in Transformers is a robust process that maps a query and a set of key-value pairs into an output, using vectors to represent the information. This output is generated as a weighted sum of the values, where a Compatibility Function between the query and the respective key in each pair determines the weights. Vaswani et al. [VSP$^+$17] introduced the Scaled Dot-Product Attention, which involves calculating and scaling the Dot Product of queries and keys before applying a Softmax Function to obtain the weights for the values. These weights and the key-value pairs' values are then utilized for another Dot Product computation.

The Multi-Head Attention concept was developed to enhance learning capabilities, enabling multiple Attention functions to operate simultaneously. The enhancement is achieved by projecting the queries, keys, and values linearly with previously learned projections and running the Attention function in parallel on all projected versions of the queries. The resulting outputs are combined and projected to create the final output. This method allows the model to learn from different representation subspaces at various positions. In Transformer models, Attention operates in three distinct positions: within the Encoder and Decoder structure, inside the Encoder itself, and within the Decoder [VSP$^+$17].

The advantages of Self-Attention lie in its superior parameter efficiency and adaptability when dealing with inputs of varying lengths, designed to model long-range dependencies using a fixed number of layers. Another key advantage of Self-Attention is its parallelization ability due to its steady sequential operations and maximum path length, which matches the length of Fully-connected Layers [LWLQ22].

2.5.3 Encoder and Decoder Structure in Transformers

Encoder and Decoder each use six identical layers, whereas the Encoder has two additional sublayers with every layer. In contrast, the Decoder has a third additional sublayer. The first two sublayers of both are the same, where the first is a Multi-Head Self-Attention mechanism, end the second is a simple, position-wise, Fully-connected Feedforward Network. The third sublayer of the Decoder uses the stacked Encoder output and performs Multi-Head Attention. A Residual Connection, which connects an output of a layer to the input of another subsequent layer, and Layer Normalization, meaning all neurons in a particular layer effectively have the same distribution across all features for a given input, are employed on all of these sublayers. To ensure the predictions can only be made on previously encountered outputs, the Self-Attention sublayer during decoding is modified to prevent positions from attending to subsequent positions. Since the Decoder can only work with previous representations, the first Transformer is a unidirectional network [VSP+17]. The Encoder itself is bidirectional, while the Decoder is unidirectional [GCMK20].

There are three ways to incorporate the Encoder and Decoder structure in Transformers. Combining Encoder and Decoder is helpful for Sequence-to-Sequence Modeling, like in Neural Machine Translation. An Encoder-only Architecture is often used for Natural Language Understanding. In this case, the outputs of the Encoder serve as a representation of the input sequence. Finally, the Encoder and the Encoder-Decoder Cross-Attention modules are removed in a Decoder-only architecture. This removal makes a Transformer suitable for Sequence Generation like Language Modeling [LWLQ22].

2.5.4 The Evolution of Transformer Models

As Transformers emerged as the top choice in Natural Language Processing, numerous modifications and variations were introduced to enhance their architecture and design. These improvements addressed a range of issues, such as the efficiency of handling long sequences, which were previously impaired by the computational and memory demands of the Self-Attention module. In addition, Generalization was boosted, overcoming difficulties in training with limited data, as the original Transformer architecture lacked assumptions on the structural bias of input data. Later iterations of Transformers were tailored to various downstream tasks and applications, making them versatile in fields like Natural Language Processing, Computer Vision, and Speech Processing [LWLQ22]. For Propaganda Detection, Transformers' Natural Language Processing capabilities are most valuable.

2.5.5 Overview of Transformers-based Language Models

Transformers have advanced significantly, demonstrating superiority over traditional Recurrent and Convolutional Neural Networks. Early Transformer models such as Generative Pre-trained Transformer (GPT) [RNSS18] and Generative Pre-trained Transformer 2 (GPT-2) [RWC+19] were limited to generating outputs based on prior context due to their unidirectional Decoder within the Encoder-Decoder structure [VSP+17]. To overcome this limitation, bidirectional networks emerged, discarding the unidirectional Decoder and focusing on a bidirectional Encoder. Bidirectional Encoder Representations from Transformers (BERT) [DCLT19] learns from a sentence's past and future context. XLNet [YDY+19] adopted BERT 's Auto-Regressive Network Architecture and bidirectional Encoder. Zhuang et al. [ZWYJ21] proposed three enhancements to BERT, resulting in Robustly Optimized BERT Pre-training Approach (RoBERTa), which was trained on a larger dataset with increased batch size and extended pretraining. However, the larger model required substantial resources for training. Subsequent models, such as ALBERT [LCG+19] and DistilBERT [SDCW19], reduced model size by factorizing Embeddings and implementing Cross-layer Sharing or using Knowledge Distillation, respectively, to create smaller yet powerful networks [GCMK20].

2.5.6 Historical Important Models

The chosen historical models marked the beginning and first advances in Transformer-based Language Models.

BERT

BERT is a Transformer-based Language Model released by Devlin et al. [DCLT19]. Besides previous architectures, BERT is designed to pretrain deep bidirectional representations from the unlabeled text by utilizing all layers' left and right input context. Pretraining is done only once during model creation. The pretrained model can be customized for various tasks by adding one fine-tuned Output Layer. A labeled dataset is used to fine-tune the pretrained BERT model for a specific downstream task during Fine-tuning. Even though the downstream tasks may differ, the base, such parameters, and data used are always the same [DCLT19]. The input representation can represent a single sentence and a pair of sentences in one input sequence. The vocabulary of BERT spans 30000 tokens. The first token of the input sequence is a unique Classification Token, which contains the whole aggregated representation of the sequence from the previous and actual layers. This special token is refered as *[CLS]* in BERT. Another special token used to separate the two distinct input sentences in pair of sentences scenario is the Separator Token: *[SEP]*. Further, a marker is added to every token to indicate belonging to the first or second sentence. An input representation is constructed by summing a Token-, Segment-, and Position Embedding for every token. During Pretraining, BERT uses Masked Language Modeling and Next Sentence Prediction. In Masked Language Modeling, a random sample of tokens on the input sequence is replaced with another special token: The Mask Token *[MASK]*. From the input tokens, 15% are selected for possible replacement, of which 80% are replaced, 10% are not changed, and a random vocabulary token replaces the left-over 10%. The Next Sentence Prediction predicts whether two segments follow each other in the original text. Its main objective is to improve performance on downstream tasks, which require knowledge about the relationship between pairs of sentences [DCLT19].

RoBERTa

After the success of BERT, a robustly optimized version was created by Zhuang et al. [ZWYJ21]: RoBERTa. RoBERTa is an extension of BERT with changes to its Pretraining phase. The model was trained longer, with bigger batches and more data. The Sentence Prediction module was removed entirely, and the sequence length was enlarged to a maximum of 512 tokens. No randomly short sequences were injected during training, and the model was trained with full-length sequences. During training, Dynamic Masking was introduced. BERT was pretrained with Static Masking: A mask was generated during Preprocessing and used for every training epoch. Zhuang et al. [ZWYJ21] changed this by generating a mask every time a new sequence is fed to the model. Due to these changes, the newly created RoBERTa model outperforms BERT in various tasks [ZWYJ21].

Cross Language Models (XLMs)

Low-resource languages typically need more substantial labeled and unlabeled data. By exploiting the capabilities of Multi-lingual Models during the Fine-tuning process, labeled data from multiple languages can be utilized to enhance performance in downstream tasks. Conneau et al. [CKG+20] developed Cross Language Model RoBERTa (XLM-R), a RoBERTa-based Cross-Language Model. Unlike other Multi-lingual Models, XLM-Rs benefits from more extensive training data and covers a larger range of languages, including those with limited resources.

2.5.7 Advanced Transformer-based Language Models

Since much progress was made in Artificial Intelligence and Transformer-based Language Models, this chapter will look at the current prevailing models to give an overview of the current state of the art. While the list of models is growing at scale, the below-mentioned models are by no means meant to be complete. The models were chosen due to public interest, like those developed by OpenAI[1], or by their direct relevance to the research community, like the open-sourced models by Meta[2].

Generative Pre-trained Transformer 3 (GPT-3)

2020, GPT-3, an Auto-Regressive Language Model, stands out with its remarkable configuration of 175 billion parameters - a tenfold increase compared to the previous Language Models. Its performance was examined in a Few-shot Setting, absent any need for gradient updates or Fine-tuning. All tasks and Few-shot demonstrations were conveyed to GPT-3 solely through textual interaction. The model demonstrated its capabilities across various datasets, covering translation and question-answering. Furthermore, it successfully handled tasks requiring on-the-fly reasoning or domain adaptation, like word unscrambling, use of an unfamiliar word in a sentence, or carrying out 3-digit arithmetic.

[1] https://openai.com (last accessed: 10 May 2023)
[2] https://ai.facebook.com (last accessed: 10 May 2023)

GPT-3 can generate news article samples that human evaluators struggle to differentiate from human-written pieces [BMR+20].

Generative Pre-trained Transformer 4 (GPT-4)

In 2023, OpenAI released GPT-4, a novel Multi-modal Model that consumes image and text inputs to yield text outputs. A central goal of this model is to boost its ability to interpret and generate natural language text, more so in scenarios requiring greater complexity and nuance. Evaluations designed for humans were employed to assess GPT-4's capabilities and manifested exceptional performance, often surpassing many human test-takers. For instance, on a mock bar exam, GPT-4's score featured in the top 10% bracket, contrasting with GPT-3, which languished in the lowest 10% [Ope23]. The bar exam is a professional test that law school graduates must pass to practice law in a specific jurisdiction or state in the USA.

The GPT-4 developers emphasized the formulation of a Deep Learning stack that scaled predictably - a necessity given the nonviable nature of comprehensive model-specific tuning for substantial training runs. The research team addressed this by innovating infrastructure and optimization strategies that displayed uniform behavior across varied scales. These advancements facilitated the reliable prediction of some facets of GPT-4's performance using smaller models trained with computational resources ranging from 1,000 to 10,000 times less [Ope23].

Open Pre-trained Transformer (OPT)

OPT was developed with a clear objective - to promote reproducible and responsible research at scale and to facilitate broader participation in the discourse on the impacts of these Large Language Models (LLMs). It is vital to have a community-wide understanding of risk, harm, bias, and toxicity in LLMs, which can only be achieved when these models are accessible for scientific research [ZRG+22].

LLMs, often trained over hundreds of thousands of computing days, exhibit extraordinary Zero- and Few-shot Learning skills. However, given their considerable computational cost, these models remain challenging to reproduce without significant financial resources. Furthermore, in the few instances where these models are accessible via Application Programming Interfaces, access to the entire model weights is typically denied, complicating their analysis. This restricted access has curtailed researchers' ability to explore how and why these LLMs function, thereby impeding their comprehension [ZRG+22].

A suite of Decoder-only pretrained Transformers, OPT, has been developed in response to this issue. These Transformers, ranging from 125 million to 175 billion parameters, are fully and responsibly shared with researchers. OPT models have been designed to align with the performance and sizes of the GPT-3 class of models, incorporating the latest best practices in Data Collection and efficient training [ZRG+22]. It has been demonstrated that OPT-175B is on par with GPT-3. Nevertheless, it requires only a seventh of the carbon footprint to develop, demonstrating its efficiency and sustainability [ZRG+22].

Large Language Model Meta AI (LLaMA)

The previous models were trained based on the hypothesis that increased parameters will correspondingly enhance performance. However, recent research has shown that the largest models may not perform best, but smaller models trained on more data [HBM+22]. LLaMA is a sequence of LLMs optimized for optimal performance across varying Inference budgets by training on more tokens than is typically used. The developed models range from 7 billion to 65 billion parameters and demonstrate competitive performance compared to other leading LLMs. Notably, despite being ten times smaller, LLaMA-13B surpasses GPT-3 in performance on most benchmarks. The model can run on a single Graphics Processing Unit (GPU) and helps researchers explore the capabilities, biases, and limitations of such Large Language Models. At the higher end of the scale, the LLaMA-65B model remains competitive with the top-tier LLMs [TLI+23].

2.6 Generating Custom Embeddings

The chapter gives an overview over different strategies that can be used to enrich Transformer-based Language Model Embeddings.

2.6.1 Embeddings from Language Models (ELMo)

ELMo offers a dynamic approach to word representation, capturing the complex aspects of word usage, such as syntax and semantics, and embracing variations in linguistic contexts, essentially addressing multiple possible meanings for a word. By utilizing bidirectional Deep Learning Models, word vectors are derived from the internal states of these models, pretrained on extensive text corpora to ensure richness in capturing the essence of words [PNI+18].

When a word vector spans a whole sentence, it is called a Sentence Embedding. In Transformer-based Language Models, where multiple Hidden Layers are employed, the output of the last Hidden Layer is often called the Aggregated Sentence Embedding for a Classification Task [DCLT19]. Another approach is to take the Embeddings from all Hidden Layers and average the values to gain a single Averaged Embedding.

2.6.2 Bag-of-Words

The Bag-of-Words technique provides a simple yet effective way to analyze text. It collects words from a text into a set, or *bag*, paying attention to their frequency but not their order or sentence structure. This process often involves removing stopwords. The advantage of Bag-of-Words is its simplicity and no need for linguistic knowledge. However, its limitation lies in its inability to capture more complex aspects like sentence structure or semantic context [MCG16].

2.6.3 TF-IDF

A standard method used in Text Classification is the TF-IDF approach. TF-IDF assigns a weight to each word in a document based on its uniqueness, effectively capturing the relevance of words within the text sequence and its corresponding category. This relevance means that words frequently used in a specific text sequence but less common in other sequences will have a higher TF-IDF score, indicating their importance and relevance to that particular sequence [YtLYc05].

2.6.4 Part-of-Speech Tags

Part-of-Speech Tagging assigns each word in a given text a label that denotes its grammatical role, whether a noun, verb, adverb, or preposition. These tags can be beneficial in specifying particular items in regular expressions, such as proper nouns for names. They can also assist in clarifying words with multiple potential tags. For instance, *book* can be a noun or a verb. However, in the sentence *book a flight*, *book* should be tagged as a verb because a noun would not be grammatically correct before a determiner and another noun [MCG16].

2.6.5 Named Entity Recognition

Named Entity Recognition is a subtask of Information Extraction that identifies entities such as individuals, organizations, dates, and locations within a text. Named Entity Recognition aims to extract specific information from unstructured text and convert it into a structured, machine-readable format. This process is crucial for building summaries, knowledge bases, and ontologies, providing a structured representation of the data [MCG16].

2.7 Deep Learning Strategies

This chapter gives an overview of different Deep Learning terms.

2.7.1 Supervised Learning

A common form of Deep Learning is Supervised Learning. During the first step, a large dataset is gathered. This dataset must represent a real-world case from which the machine can learn. The input data is then categorized into different categories called labels. Those labels form the desired output made by the Deep Learning system. The Deep Learning system is shown the input data and its label during learning. The machine must then conclude how the input data can be used to predict the label. The machine generates and refines its outputted Feature Vector with every learning step. In an ideal setting, the final refined output vector would look precisely like the vector representing the to-be-predicted label. In this case, the machine is sure to predict the desired category. The Deep Learning system calculates a score for every category available. The category with the highest score is then predicted by the algorithm as its supposed guess. The score is calculated by a function that measures the difference between the predicted and actual labels. The distance can be calculated since both are represented in abstract vectors. Once the difference between the two vectors has been calculated, the Deep Learning algorithm changes its internal parameter to reduce the error. Those parameters are called weights, and hundreds of millions may be in a single Deep Learning system [LBH15].

The weights are adjusted by calculating a Gradient Vector. This vector shows how a slight change in the weight would change the difference between the true and the predicted value. The overall goal of a Deep Learning system is to reduce the error until no further optimization is possible and to make the error as small as possible. A strategy called Stochastic Gradient Descent is utilized to tackle the Minimization Problem in practice. During Stochastic Gradient Descent, the input vector is checked on a few examples. Then, an output is generated with the resulting error terms. The Average Gradient is calculated for those few examples, and the weights are adjusted accordingly. This calculation is done as long as the average of the objective function stops decreasing. Finally, after the model has been trained, a separate part of the training data is used to test the model's ability to predict the correct label on unseen data [LBH15].

2.7.2 Semi-Supervised Learning

Unfortunately, it is often difficult or expensive to collect and label training data for Supervised Learning [WKW16]. Semi-Supervised Learning combines Supervised and Unsupervised Learning [CCZ06] [Zhu08]. Often Semi-Supervised Learning is utilized to improve a model in either a Supervised or Unsupervised Learning environment. Solving a Classification Problem in Supervised Learning might require additional data points to improve the classifier's output. Semi-Supervised Learning has also been applied to areas where labeled data exist. However, the unlabeled data brings new helpful information for prediction so that the performance can be improved [vEH20].

2.7.3 Unsupervised Learning

During Supervised Learning, a set of data points consisting of some input data and its corresponding output value is provided. In Unsupervised Learning, no specific output value is provided. Instead, the model tries to find an underlying structure from the input data [WKW16]. Unlabeled data is used to train a Deep Learning Algorithm based on its encountered features from the dataset.

2.7.4 Self Training

Self Training uses a single supervised classifier that has been iterative trained on labeled data and pseudo-labeled data [vEH20]. It is called pseudo-labeled data since an insufficient model categorizes unlabeled data. The most confident predictions are added to the original data, and the supervised classifier is trained on the new and original data from before until no more unlabeled data remains. The procedure of Self Training requires different design decisions, such as the selection of data to be pseudo-labeled and the re-usage of this data for learning [vEH20]. The pseudo-labeled data is sometimes referred to as *Silver Data.*

2.7.5 Minority Classifier

A Minority Classifier refers to a Machine Learning model trained to identify and categorize instances of the Minority Class in an imbalanced dataset. Imbalanced data is a common problem in Machine Learning, where unequal classes exist. The Minority Class has fewer instances and is often the more important class in problems such as Fraud Detection or Disease Diagnosis [NS16].

2.7.6 Hierarchical Classification

Hierarchical Classification is a Classification Problem where the classes are organized into a hierarchical structure. Unlike other Classifications, where each class is treated independently, Hierarchical Classification acknowledges and utilizes the relationship between classes. This type of Classification is beneficial in scenarios where classes naturally form a hierarchy [SF11].

2.7.7 One-Versus-One Classifier

A One-vs-One Classifier is a method for creating an individual classifier for each category pair. If there are three classes (Class 1, Class 2, Class 3), a classifier for Class 1 vs. Class 2, another for Class 1 vs. Class 3, and finally, one for Class 2 vs. Class 3 is created. When determining the class of a new item, the item is run through all the classifiers. Each classifier votes on what they perceive the item to be, and the class that accumulates the most votes is the winner. An excessive number of classes can result in a large number of classifiers. However, the method often delivers more accurate results than other methods. The alternative is a One-versus-All classifier, where only one classifier is trained, that calculates the probabilities for one item being part of the given classes [MG13].

2.7.8 Single- vs. Multi-class Classification

Single-class Classification is a Machine Learning problem where each example is categorized into one of two classes. On the other hand, Multi-class Classification is when each item can belong to multiple predefined categories. Both types of classification aim to build a learning model from labeled training data that can accurately predict the category of new, unlabeled objects [MG13]. Deriving from this, in a Multi-label Multi-class Classification scenario, each example is not restricted to a single label but can be associated with several labels out of many potential ones. Simply put, classes in this context are not exclusive, and an example can be classified under multiple categories simultaneously.

2.7.9 Transfer Learning

High-performing models can be created by utilizing data from different domains. This technique is called Transfer Learning, where information from one domain can improve a learner from a related but different domain. Many domains seem distinct, but when looking at their high-level domain, they often share specific characteristics [WKW16].

2.8 Additional Topics

This chapter explains further relevant topics in the context of the thesis.

2.8.1 Ridge Regression

In ordinary Linear Regression, a line is sought that minimizes the sum of the squared differences between the actual and predicted values. However, when predictor variables are highly correlated, this can lead to problems. The model becomes unstable, meaning small changes in the data can lead to significant changes in the predictions. Also, the model can overfit, meaning it is too closely tailored to the training data and needs to perform better on new data. Ridge Regression addresses this by adding a penalty term to the equation that the model is trying to minimize [HK00].

2.8.2 Longest Common Subsequences

The Longest Common Subsequence is a concept used to compare two strings. It refers to the maximum number of identical symbols or elements found in both strings while maintaining the order of these symbols. This similarity measure is vital in many fields, including spell-checking applications, molecular biology, and file archiving systems. In each of these cases, the aim is to assess how alike two sequences are. When comparing words, the Longest Common Subsequence provides an effective metric for evaluating the degree of resemblance. The larger the Longest Common Subsequence, the closer the two strings are considered to be [BHR00].

2.8.3 Conditional Random Field

Conditional Random Field (CRF) is a technique for segmenting and labeling sequential data. First, an undirected graphical model is employed. A single Log-linear Distribution is defined over sequences of labels given a particular sequence of observations. The unique advantage of CRFs lies in their conditional nature. A label is not predicted based solely on an individual sample, but the context of surrounding samples is also considered. Dependencies between samples can be modeled as individual predictions are incorporated into the graphical model. This results in a powerful and flexible tool for dealing with sequential data [LMP01].

2.8.4 Evaluation Metrics

The performance of a Deep Learning model can be measured with the help of the most common Evaluation Metrics in the Machine Learning domain. These are Accuracy, Precision, Recall, and the F1-Score. The Recall is the True Positives Rate, which tells how many propaganda techniques were predicted correctly. Further, Recall tells us how many of the Real Positives are detected by a model. So, if it is known that a specific propaganda technique appears ten times, Recall checks this with the encountered number of the propaganda techniques. Precision denotes how many Predicted Positives are truly Real Positives. Precision tells how accurate a model is when comparing the predicted and actual propaganda techniques. Precision and Recall are insufficient to assess a model's quality in some situations. The F1-Score is an approach to balance Precision and Recall. The Harmonic Mean, named F1-Score, references the True Positives to the Arithmetic Mean of Predicted Positives and Real Positives [Pow11]. The focus during performance evaluation is on the Micro F1-Score since the challenge dataset has an uneven class distribution and a balance between Precision and Recall is needed. Also, the F1-Score was used during the challenge as the prioritized measure for assessing participant's performance [DSMBCW$^+$20].

CHAPTER 3

Literature Review

This chapter hosts the review of important literature, describes the search process and finally the analysis of all submissions of the SemEval 2020 Task 11 challenge.

3.1 Reviewing the Literature

The Literature Review is an essential part of the research process. It helps to find important academic sources that add depth to scientific work and makes it easier to understand the specialized terms and concepts used. It is also helpful in creating a list of relevant sources for the research topic. After searching for literature, the next step is to review the found resources and determine their relevance to the research [RS04].

3.2 Searching the Literature

The first step is to search Google Scholar[1] to find all the papers related to the *SemEval 2020 task 11* challenge. The search uses the specific phrase *intitle:"semeval 2020 task 11"*. This phrase helps filter out papers that include this exact phrase since all submissions for this challenge included *semeval 2020 task 11* in their title. The search results show 33 papers, which is close to the number of participants mentioned by Da San Martino et al. [DSMBCW+20].

The final step performed was a Backward Search to make sure no relevant publications had been missed by checking the mentioned participants by Da San Martino et al. [DSMBCW+20].

[1] https://scholar.google.com (last accessed: 22 December 2022)

3.3 Analysis of the SemEval-Challenge

After the challenge, 25 teams published a paper about their approach to detecting propaganda. The following chapter looks at every submission and categorizes the different approaches into categories: Preprocessing, Model Choice, System Setup, Postprocessing, and Ensembling Strategy. These categories will later be used to build mixed approaches from all the ideas the challenge participants incorporated. The approaches of the first five submissions will be analyzed in greater detail, while further submissions will be scanned for ideas and overall approaches.

3.3.1 Overall Analysis of Model Choice

The participants used different models, with a clear trend to using the Transformer architecture, more precise Transformer-based Language Models.

Jurkiewicz et al. [JBKG20], Chernyavskiy et al. [CIN20], Raj et al. [RJR$^+$20], Singh et al. [SSKM20], Grigorev and Ivanov [GI20] used only RoBERTa as their preferred Learning Model. Jurkiewicz et al. [JBKG20] do not state if they used a cased or uncased model, while Chernyavskiy et al. [CIN20] proposed the usage of a cased model.

Morio et al. [MMOM20] built an Ensemble Model consisting of BERT, RoBERTa, XLNet, XLM, Albert, and GPT-2, Kim and Bethard [KB20] also decided to use multiple models, namely BERT and RoBERTa.

Most of the challenge participants stick with BERT as their preferred Transformer model, like Dimov et al. [DKS20], Blaschke et al. [BKT20], Bairaktaris et al. [BSA20], Kaas et al. [KTP20], Krishnamurthy et al. [KGY20], Altiti et al. [AAO20], Jiang et al. [JGM20], Patil et al. [PSA20], Paraschiv and Cercel [PCD20], Li and Xiao [LX20], Daval-Frerot and Weis [DFW20], Dao et al. [DWZ20] and Kranzlein et al. [KBG20].

Additionally, Kaas et al. [KTP20], Patil et al. [PSA20], and Li and Xiao [LX20] stacked BERT with a Logistic Regressor, while Dao et al. [DWZ20] and Kranzlein et al. [KBG20] combined BERT with a LSTM layer.

Only four submissions decided to go without a Transformer architecture. Petee and Palmer [PP20] used a Logistic Regressor, Ermurachi and Gifu [EG20] tested a Random Forest approach, Arsenos and Siolas [AS20] still relied on Deep Learning, using a Multilayer Perceptron with a LSTM layer, as like Martinkovic et al. [MPS20] relying on LSTM in their submission.

3.3.2 Analysis of Best Five Submissions

Beginning with the best five submissions the different approaches in terms of Preprocessing, System Setup, Post Processing, and Ensemble are summarized.

Preprocessing

In Machine Learning, the input data is one of the keys to build an excellent predicting model. Therefore most of the challenge participants put a big emphasis on Preprocessing the given challenge dataset in different ways.

The first placed Jurkiewicz et al. [JBKG20] tested two approaches, of which the first was to use the left and right context of the propaganda span as the input, and the second to insert special tokens around the span, which should be classified as a propaganda class. The second approach leveraged the context again, this time without a special token around the propagandistic span. The context had the maximum possible size of 512 words. Therefore, the left context includes 256 subwords, like the right context.

The research by Chernyavskiy et al. [CIN20] reduces the Multi-label Multi-class Problem to a Single-Label Multi-class Problem by creating copies of spans with multiple labels. The construction of the input span involves a combination of the propaganda span with the corresponding sentence. This combination is divided by a Separator Token and initiated with a Classification Token.

A proposal in which no context was utilized was put forward by Morio et al. [MMOM20]. In their approach, the Sentence Embedding of the propaganda span was fused with Part-of-Speech Tags and TF-IDF Tags.

The study by Raj et al. [RJR^{+}20] involved experimenting with three different ways of feeding data into their RoBERTa model. The initial method used only the propaganda span itself, while the second method used only the context of the span. Lastly, a combination of both the propaganda span and context was considered. The final chosen approach involved this combination of both elements.

Again, the span context was utilized in the research by Singh et al. [SSKM20]. Rather than merging the propaganda span with the surrounding context, these elements were individually introduced into two pretrained Transformers. The context was restricted to a total of 130 characters. The sequence representations from each model were retrieved and combined, resulting in a final Embedding that combines the contextualized outputs of both models.

System Setup

Three unique systems were utilized in the study by Jurkiewicz et al. [JBKG20]. The initial system made use of RoBERTa in both of its variants. The first variant ended at this stage, whereas the second added a small stacked Transformer on top of RoBERTa, which solely used the propaganda span's Embedding. This added Transformer was characterized by three Hidden Layers, four Attention Heads, and an Intermediate Layer of size 512. The second system was introduced in response to the significant imbalance in the dataset. Here, weights dependent on each class were utilized. These weights were computed by determining the Inverse Frequency with the frequency of the most prevalent class. Subsequently, these weights were incorporated into the loss function.

The final system, meanwhile, integrated the components of the first and second systems, complementing them with Self Training. No high-confidence examples were chosen, and no loss correction was done for noisy annotations. The top-performing model from the Span Identification task was repurposed to annotate 500,000 random sentences from OpenWebText[2]. As a result, the authors identified more sentences with propagandistic elements, which they then reused to annotate an additional dataset.

In their research, Chernyavskiy et al. [CIN20] developed two model versions. The first version used the input span, and the Sentence Embedding was obtained from the Classification Token. This token was passed through an additional Softmax Layer to generate a prediction. The second version of the model combined the extracted Classification Token from the Sentence Embedding with the Averaged Embeddings from the remaining propaganda span and the span length. Each Hidden Layer of a Transformer-based Language Model produces an Embedding for the propaganda span. As the process progresses, the sentence representation becomes increasingly accurate. An Averaged Embedding is created by calculating the mean of the Embeddings of all Hidden Layers for additional learning. A Fully-connected Layer was added on top of this. Furthermore, Transfer Learning was employed as a third strategy. The model was initially trained using data from the Span Identification subtask, and then further training was done during Technique Classification.

In the approach by Morio et al. [MMOM20], two separate Feedforward Networks are utilized, into which the concatenated input is introduced. The role of the first Feedforward-Network is to procure the sentence representation, whereas the second one is employed to achieve representation from all tokens within the propaganda span. The final model input is assembled by merging the sentence representation, the representations of tokens found at the start and end of the propaganda span, and finally, the representations garnered through Attention and Pooling. An additional label-wise Feedforward Network and a Linear Layer are incorporated to extract information specific to each propaganda technique. Furthermore, weights corresponding to the proportion of positive samples are allocated to the loss function to manage class imbalance. During Inference, labels predicted for each sentence are arranged in descending order and assigned to labels according to their order in a multi-label span.

In the study by Raj et al. [RJR$^+$20], three interconnected systems were constructed to classify the propaganda techniques. The first system adapts the RoBERTa model, modifying the final layer, resulting in 14 Hidden Layers, with the last being a Softmax Layer. This last layer is trained on the downstream Technique Classification task, while the other layers are fine-tuned beforehand. Due to the high imbalance in the dataset, a system termed the *Minority Classifier* is introduced. It comprises five separate Hierarchical Classifiers, termed as level-1 classifiers, focusing on the five Minority Classes. Each level-1 classifier is an Ensemble of 13 One-versus-One Classifiers, named level-2 classifiers. The outputs of these level-2 classifiers are collated to procure the prediction

[2] https://github.com/jcpeterson/openwebtext (last accessed: 23 December 2022)

for the level-1 classifiers. If the prediction confidence surpasses a certain threshold, the span is considered a positive example of the Minority Class. The level-2 classifiers are straightforward Linear Classifiers, aiding in faster computations during learning. The *Repetition* class was also managed separately, but not during Postprocessing like by Chernyavskiy et al. [CIN20] but during the training phase. This issue identifies the presence of the Longest Common Subsequences between the propaganda span and context. Rather than employing an exact match approach, they calculate the presence of *Repetition* by determining a percent match between the span and context, with a threshold that adjusts based on the length of the fragment.

After obtaining the concatenated Embedding, Singh et al. [SSKM20] passed it to a Classification Layer on top, which performs Technique Classification. However, before this, an additional Hidden Layer reduces the dimension of the Context Embedding. Since the reduced dimensions, the additional classifier gives more attention to the actual propaganda span.

Postprocessing

Chernyavskiy et al. [CIN20] experienced some difficulties with their first models, mainly when predicting the *Repetition* technique. This issue was addressed during the Postprocessing stage, where the presence of any given span was examined throughout the entire dataset. This process involved looking for exact matches; once punctuation was removed, stopwords were filtered out, and stemming was applied. The label for *Repetition* was assigned if the span in question matched at least two other spans. However, if only a single match existed, the classifier had to predict the label with a minimum threshold of 0.001. If no match was found, the probability was set to zero unless the classifier had predicted the label with a minimum probability of 0.99.

Ensemble

In their work, Jurkiewicz et al. [JBKG20] utilized an Ensemble method that averaged the class probabilities from their three developed systems, each trained with different Hyperparameters.

Similarly, Chernyavskiy et al. [CIN20] employed an Ensemble approach, combining several model variations to construct a more robust final model. However, their paper did not resolve the specific details of this process.

An Ensemble strategy based on Stacked Generalization was adopted by Morio et al. [MMOM20]. This method used multiple classifier predictions as inputs for a Meta Estimator. Hyperparameter Search and Cross-Validation were performed to optimize the Meta Estimator, with the Learning Rate and Dropout Ratio being the key Hyperparameters. The final number of models generated was determined by multiplying the number of pretrained Transformer models by the number of Hyperparameter sets and K-folds. Only those with the best validation scores were selected among these models for further processing. These selected models then predicted their validation folds on the training data.

The predicted validation folds were concatenated for each Hyperparameter set, creating meta-features to train the Meta Estimator. The labels were predicted during testing using fine-tuned models with the best Hyperparameters. The trained Meta Estimator used these predicted labels to yield the final prediction.

Lastly, Raj et al. [RJR$^+$20] implemented an Ensemble strategy at various stages of their model. The level-2 and level-1 classifiers were used in Ensembles to generate the final prediction.

3.3.3 Analysis of the Rest

After looking deeply at the best five submissions, all other submissions are analyzed in this section.

Preprocessing

Grigorev and Ivanov [GI20] implemented Undersampling for the over-represented classes *Loaded Language* and *Name Calling, Labeling*, setting Undersampling ratios of 0.2 and 0.5, respectively.

In their Preprocessing phase, Blaschke et al. [BKT20] extracted Named Entity Tags, including predictions for nationalities, religious or political groups, and geographic entities. They introduced question features and rhetorical question features.

Bairaktaris et al. [BSA20] also used Preprocessing techniques to create four lists containing countries, politically related words, religions, and slogans. During Preprocessing, propaganda spans were scanned for list items replaced with corresponding tags. Named Entity Recognition was applied, replacing politicians' names with a *PERSON* tag providing the best results.

Kaas et al. [KTP20] incorporated 54 hand-crafted features into their model, identifying five as particularly performance-driving. These included counting the reappearance of stemmed one-word spans and spans longer than one word, counting the number of words in a span, and calculating the inverse uniqueness of words in a span.

Krishnamurthy et al. [KGY20] extended BERT with Emotional Intensity Analysis of propaganda spans and extracted 73 word-level psycholinguistic features from the Linguistic Inquiry and Word Count (LIWC) lexicon [PBJB15].

Kim and Bethard [KB20] experimented with the span context by modeling inputs with only the propaganda span, the parent sentence, and the sentences before and after the propaganda span.

Next, Altiti et al. [AAO20] preprocessed the spans by removing punctuation and special symbols, performing Tokenization, and cleaning contractions.

Martinkovic et al. [MPS20] reduced the number of input tokens by removing Tweet footers, timestamps, web surveys, hyperlinks, advertisements, emoji characters, Twitter[3] mentions, and substituting unicode quotation marks, apostrophes, and hyphens with their ASCII equivalents.

Jiang et al. [JGM20] lowercased and tokenized the propaganda spans for their BERT-based model. To tackle the imbalanced dataset, they undersampled Majority Classes and oversampled Minority Classes to 400 examples per class.

In their Preprocessing phase, Patil et al. [PSA20] , removed non-ASCII characters, performed UTF-8 conversion, lowercasing, stemming, and removed trailing white spaces, newlines, and stopwords. Their feature extraction process involved extracting contextual, content-, and context-based metadata.

Different text-splitting approaches were tested by Paraschiv et al. [PCD20]. However, no approach was able to better the results. Therefore only the exact propaganda span was used as input for their BERT-based model. The authors used Masked Language Modeling on two corpora to further train the BERT model on propagandistic spans.

Paraschiv et al. [PCD20] used only the exact propaganda span as input for their BERT-based model, which was further trained on two corpora containing 8.5 Million Fake News articles[4] and 750000 articles from the Hyperpartisan news corpus [KMS$^+$19].

Li and Xiao [LX20] employed an emotion lexicon to extract word intensity, which was then added to the BERT Embedding. Class weights were introduced to manage class imbalance.

Daval-Frerot and Weis [DFW20] described three approaches to gather more data for the Technique Classification subtask: Backtranslation, Synonym Replacement, and TF-IDF. For their model, they did not use Backtranslation of the lack of a proper translation package. For Transfer Learning, the All-the-news articles[5] dataset was used.

About 3000 additional training samples were created by Kranzlein et al. [KBG20] by randomly replacing verbs, nouns, and adjectives with synonyms. Part-of-Speech Tags, Named Entity Tags, and Keyword Frequency were used as hand-crafted features.

Arsenos and Siolas [AS20] performed minor text alterations by removing most punctuation marks.

Ermurachi and Gifu [EG20] eliminated stopwords and special characters, cleaned initial and ending white spaces, and lowercased the text. They also used Bag-of-Words and TF-IDF during Feature Engineering.

Petee and Palmer [PP20] introduced a primary classifier for a specific task using two features: Bag-of-Words and the average of all the words in the section. Then more

[3] https://twitter.com (last accessed: 03 June 2023)

[4] https://github.com/several27/FakeNewsCorpus (last accessed: 04 January 2023)

[5] https://components.one/datasets/all-the-news-articles-dataset (last accessed: 05 January 2023)

features were added, including information about Named Entities and their sentiment across three dimensions: valence, arousal, and dominance.

Finally, Verma et al. [VMC20] used character-level annotations to create word-level inputs, kept track of the character-level indices of each word, removed empty sentences, and combine sentences that were part of the same propaganda span. For their BERT and ELMo-based models, they formatted the data as a sequence of input and output and trained their model by One-hot Encoding the tokenized words. They generated sentence representations using a pretrained Word Embedding with LSTM and BERT.

System Setup

The Cost-Sensitive Learning approach, utilized by Grigorev and Ivanov [GI20], addressed class imbalance by assigning inverse weights relative to the specific class proportion in the dataset. This approach resulted in non-zero F1 scores across all classes.

The system proposed by Blaschke et al. [BKT20] primarily focused on the *Repetition* class. It was achieved by breaking down the prediction generation into subsystems, including *Base* and *Repetition Models*. A Multilayer Perceptron provided the final prediction. If both *Base* and *Repetition Models* failed to predict the *Repetition* class, a third model would reclassify the fragment into one of the remaining 13 classes.

Implementing BERT outperformed the traditional Machine Learning proposed by Bairaktaris et al. [BSA20]. The most effective approach involved labeling the data with *NATION*, *RELIGION*, *POLITICS*, or *SLOGANS*, which yielded improved results, especially in some Minority Classes.

The complete model proposed by Kaas et al. [KTP20] consists of three building blocks. The first building block is a BERT model, where input was only the actual propaganda span without context. A 10-fold Stratified Learning Strategy was used to obtain the best model. Ten stratified splits were created from training data. The fine-tuned model was fed with the folds until no further loss increase was encountered on the previously created test split from the training data. A Linear Layer was modeled on top to get a 14-dimensional output vector representing the 14 classes. The second model was a Logistic Regression model used with the extracted features. The result of this model was then concatenated with the output of the BERT model and the extracted features. The last component, a Fully-connected Feedforward Network with three Hidden Layers, took the output from BERT, the hand-crafted feature, and the output of the Logistic Regression as its input and returned the final prediction.

In the approach by Krishnamurthy et al. [KGY20], various feature sets were concatenated and processed through a Dense Layer. The output layer of the system was a Fully-connected Layer with Softmax Activation. Notably, including the LIWC Lexicon [PBJB15] did not improve the results, while adding information about emotions did improve the final result.

Kim and Bethard [KB20] exploited different variations of BERT and RoBERTa in their submission. The final model combined BERT-large and RoBERTa-base without a context feature.

Altiti et al. [AAO20] selected BERT as their optimal model after testing various other options, including a simple Neural Network and a Convolutional Neural Network.

The custom model submitted by Martinkovic et al. [MPS20] utilized ELMo word representations, a single Bi-LSTM Encoding Layer, a Self-Attention Layer, and a Linear Layer for decoding.

Jiang et al. [JGM20] applied Bagging on nine BERT models, each trained on different subsets of the training data. The most frequently occurring prediction was selected.

Patil et al. [PSA20] built a model that combined input spans processed through BERT and Logistic Regression models. The output from these models was then fed into another Logistic Regression model to generate the final prediction.

To pretrain BERT, Paraschiv et al. [PCD20] used two additional corpora to sensibilize BERT for linguistic structures of propagandistic spans. The corpora were used to train BERT for 2 million steps. The resulting pretrained BERT model is extended with a Dense Layer funnel of 768, 256, and 14, followed by a final Softmax Layer. If a span overlapped with another span, the two overlapping spans were not labeled with the same class. The authors assigned distinct classes in such cases.

Li and Xiao [LX20] implemented a three-part approach consisting of a BERT model with Cost-Sensitive Learning, a model for emotional features, and a Logistic Regression model with various continuous and boolean features.

Daval-Frerot and Weis [DFW20] fine-tuned a simple BERT model and compared it to a similar bidirectional LSTM network to avoid Overfitting. Their model was based on Unsupervised Data Augmentation [XDH$^+$20].

Arsenos and Siolas [AS20] trained a Multilayer Perceptron using pretrained word vectors from the Word2Vec model [MCCD13] and Word Embeddings from GloVe [PSM14]. Before being used, the data was processed through a bidirectional LSTM.

Ermurachi and Gifu [EG20] used traditional Machine Learning, specifically Bag-of-Words, and TF-IDF, to create a feature set for a Random Forest [Ho95] classifier.

Lastly, the Multi-System-Framework by Verma et al. [VMC20] utilized BERT with a weighted loss. The output was fed into three systems: a Linear Layer with Dropout, a Linear Layer with Multi-Sample Dropout, and a Convolutional Neural Network with Multi-Sample Dropout. The outputs of all three were aggregated to form the final prediction.

Postprocessing

As described in the System Setup, Blaschke et al. [BKT20] leveraged Postprocessing to overwrite the predictions from their model, if applicable. Also, the authors observed the input fragments for duplicates, and if found, a model was used to label the duplicate instance [BKT20].

After Ensembling their different learning models, Li and Xiao [LX20] used Rule-based Correction and Reinforcement to reassign classes if the *Repetition* class is predicted [LX20].

Kranzlein et al. [KBG20] used a LSTM model, in which BERT Embeddings and all features were fed in.

Ensemble

Kaas et al. [KTP20] ensemble the results of the BERT model, the extracted features, and the result of the Logistic Regression to obtain a better-performing model from two weak learners.

The Ensemble Model proposed by Kim and Bethard [KB20] is based on an Average Ensemble. The combined models were a BERT-large and a RoBERTa-base model.

A completely different approach in Ensemble Learning is used by Li and Xiao [LX20]. Instead of averaging their different models' predictions, they took the *Repetition* class from the Logistic Regression Model, the Majority Classes from the fine-tuned BERT model, and the Minority Classes from the Cost-Sensitive Learning approach.

Daval-Frerot and Weis [DFW20] repeated their training multiple times with minor variations. The results were aggregated into an Ensemble not further described in their paper.

After the BERT model, Verma et al. [VMC20] ensemble their prediction from their Linear Layer with Dropout, a Linear Layer with Multi-Sample Dropout, and a Convolutional Neural Network with Multi-Sample Dropout.

CHAPTER 4

CRISP-DM

4.1 Business Understanding

The European research on disinformation and propaganda [BHL+21] is used to elaborate a comprehensive understanding of the Business Context and its potential stakeholders involved.

4.1.1 Business Context 1: Addressing Media Distrust

Examining the role of propaganda and disinformation in fostering distrust in media information is part of the first identified Business Context. By identifying the key propaganda techniques used and making the propaganda usage visible, this thesis aims to improve media credibility, transparency, and trustworthiness. Those improvements enable the public to better distinguish between reliable sources and those that promote disinformation, fostering a more informed and engaged society [BHL+21]. Possible stakeholders could be news organizations, journalists, media regulatory bodies, social media platforms, fact-checking organizations, educational institutions, and the general public.

4.1.2 Business Context 2: Preserving Democratic Processes

Identify and analyze disinformation and propaganda content to protect democratic processes in the European Union and its member states. By understanding the techniques and strategies employed, policymakers and stakeholders can develop targeted countermeasures that minimize disinformation's negative impact on democratic processes. Possible stakeholders could be: European Union institutions, national governments, policymakers, political parties, electoral commissions, media organizations, civil society organizations, and the general public.

4.1.3 Business Context 3: Restoring Trust in Institutions

By analyzing the media coverage of the attack on Ukraine, propaganda techniques that erodes trust in institutions could be revealed. Finding propaganda in state-published media could raise awareness towards manipulation-free communication. Doing so aims to help restore citizens' confidence in these institutions, fostering a more stable and functional democratic system. Possible stakeholders are National governments, European Union institutions, policymakers, media organizations, journalists, social media platforms, civil society organizations, and the general public.

4.1.4 Business Context 4: Supporting Media Literacy

By providing a tool to identify propaganda in online news articles, this thesis contributes to media literacy initiatives. By helping users better understand and critically evaluate the information they consume, an informed and discerning public, better equipped to navigate the complex media landscape, is fostered. The possible stakeholders are media literacy organizations, educational institutions, teachers, students, researchers, media watchdog groups, social media platforms, non-governmental organizations promoting media literacy, and citizens interested in becoming more informed and the general public.

4.1.5 Conclusion on Business Understanding

In total more than four Business Contexts were identified during the analysis of the European research on disinformation and propaganda [BHL$^+$21]. However, not all were suitable to be thematized within the scope of the thesis.

4.2 Data Understanding

This chapter outlines the steps necessary to entirely understand the data used for the project, including Data Collection, Exploration, and Quality Assessment.

The data used can be split into two distinct datasets. The first dataset trains, optimizes and evaluates the Deep Learning model. After model training, the second dataset is used to analyze the news articles published before and during the Ukraine crisis.

4.2.1 Propaganda Techniques Corpus

To be able to train the Propaganda Detection Model, an annotated dataset for Supervised Learning is used.

Data Collection

The Propaganda Techniques Corpus is a dataset of texts annotated with 18 fine-grained propaganda techniques. Six professional annotators manually annotated the corpus, and specific underrepresented techniques were merged for simplification. The dataset is organized into training, development, and test sets, with articles in plain-text format. The dataset contains 446 articles from 48 news outlets. The corpus enables three tasks: Propaganda Span Identification, Propaganda Technique Labeling, and Fragment Level Classification. For each task, the propaganda labels are provided [DSMYBC$^+$19].

Data Exploration

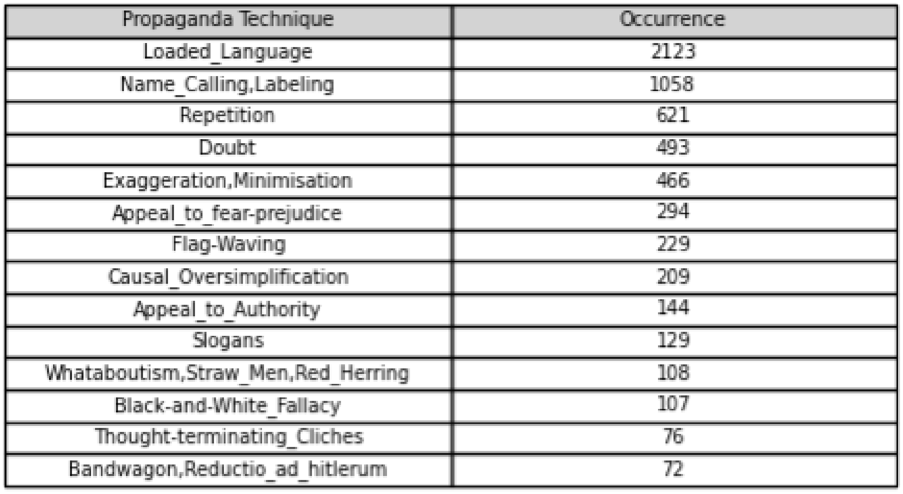
The Propaganda Technique Corpus is divided into training, development, and test sets. However, the test set is not annotated, as it was created to organize public competitions. Consequently, the test set was evaluated by submitting model inference to a website that calculated the final evaluation performance. This option was turned off after the challenge ended. For this thesis, this implies that it is not feasible to evaluate the results of the test set. The challenge summary published the Evaluation Metrics for the development set. As a result, the training set will be utilized to train the model. In contrast, the development set will be employed for evaluation and model ranking based on the challenge participants' results on the development dataset.

The training dataset comprises 6128 distinct propaganda spans, with two classes, *Loaded Language* and *Name Calling/Labeling* - being significantly overrepresented, having 2123 and 1058 instances, respectively. Additionally, *Repetition* (621 instances), *Doubt* (493 instances), and *Exaggeration/Minimization* (466 instances) are also prevalent, forming the five Majority Classes in the dataset. Conversely, there are nine Minority Classes, each accounting for less than 5% coverage. These include *Appeal to fear-prejudice* (294 instances), *Flag-Waving* (229 instances), *Causal Oversimplification* (209 instances), *Appeal to Authority* (144 instances), *Slogans* (129 instances), *Whataboutism/Straw Men/Red Herring* (108 instances), *Black-and-White Fallacy* (107 instances), *Thought-terminating Cliches* (76 instances), and *Bandwagon/Reductio ad Hitlerum* (72 instances). The findings are summarized in Figure 4.1. Next, Figure 4.2 shows the average span length of the different propaganda techniques. The shortest spans are within *Loaded Language*, *Name Calling/Labeling*, *Repetition*, *Slogans*, and *Thought-terminating Cliches*, while the longest are within the *Doubt*, *Appeal to Authority*, *Causal Oversimplification*, and *Black-and-White Fallacy* techniques.

The development dataset contains 1063 examples of propaganda with a nearly identical distribution of the occurrences of the propaganda techniques.

Data Quality Assessment

The Propaganda Technique Corpus is an imbalanced Multi-label Multi-class Dataset, which raises challenges in developing an accurate Propaganda Detection Model. However, the overall data quality is adequate, as Da San Martino et al. [DSMYBC$^+$19] have already implemented measures to maintain high quality. Despite these efforts, the articles have not been cleaned, so the text may still contain non-relevant information, such as author names or Twitter cards.



Propaganda Technique	Occurrence
Loaded_Language	2123
Name_Calling,Labeling	1058
Repetition	621
Doubt	493
Exaggeration,Minimisation	466
Appeal_to_fear-prejudice	294
Flag-Waving	229
Causal_Oversimplification	209
Appeal_to_Authority	144
Slogans	129
Whataboutism,Straw_Men,Red_Herring	108
Black-and-White_Fallacy	107
Thought-terminating_Cliches	76
Bandwagon,Reductio_ad_hitlerum	72

Figure 4.1: Propaganda Technique Distribution for Training

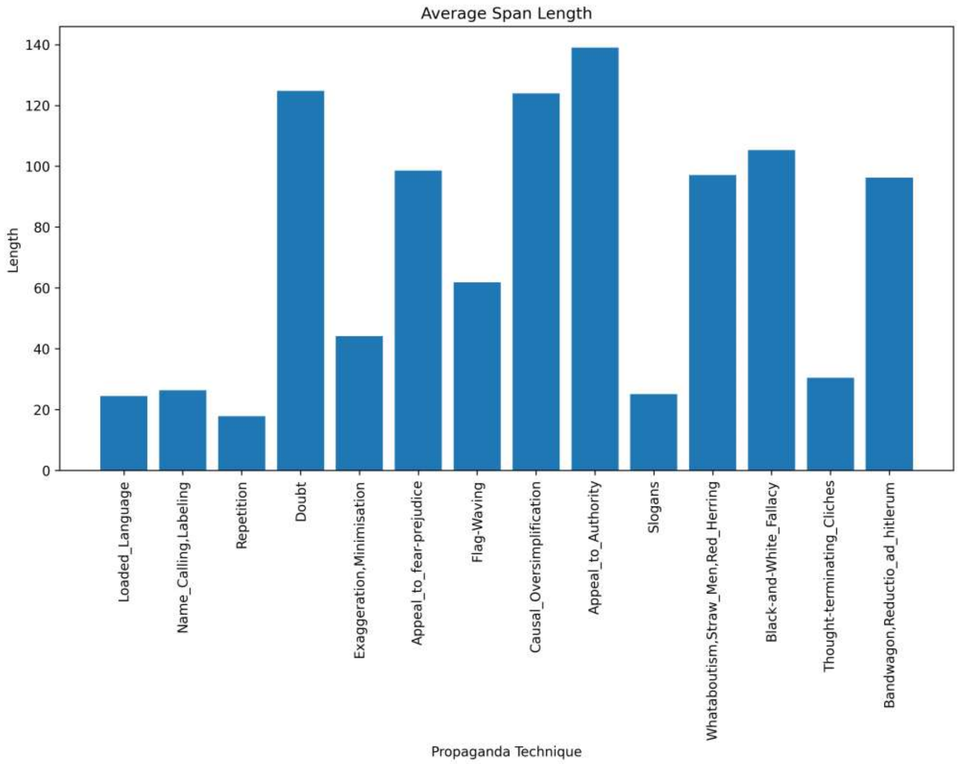


Figure 4.2: Average Span Length of Propaganda Spans for Training

4.2.2 Analysis Dataset

The Analysis Dataset is used to analyze the self-gathered articles from Russian and American sources.

Data Collection

The Analysis Dataset comprises these specific articles after developing the model to analyze propaganda usage in American and Russian news articles. The news articles were gathered from ten different sources, with five for each country. The American websites scraped include ABC News[1], CBS News[2], CNN International[3], Fox News[4], and Politico[5].

The Russian websites scraped are News Front[6], Novye Izvestia[7], RT[8], Sputnik News[9], and Tass Russian News Agency[10]. Each article's title, subtitle, body, and creation date were saved. The Data Collection ranges from 01 January 2022 to 26 May 2023. Initially, 14689 American news articles and 55231 Russian news articles were scraped.

Data Exploration

Further information on the propaganda spans will be provided in Chapter 5 while evaluating the results.

Data Quality Assessment

For Politico and News Front, the scraped articles range from 23 February 2022 to 24 April 2023. The scraping of articles before and after this range failed multiple times. Since the difference in articles was only around 200, respectively, the decision was made to accept this minor cut in Data Quality.

4.3 Data Preparation

The Data Preparation stage of the CRISP-DM methodology is essential for transforming raw data into a format suitable for analysis and modeling. This chapter outlines the critical steps in preparing the project's data, including Data Cleaning, Integration, Transformation, and Feature Engineering.

[1] https://abcnews.go.com (last accessed: 08 July 2022)
[2] https://www.cbsnews.com (last accessed: 08 July 2022)
[3] https://edition.cnn.com (last accessed: 08 July 2022)
[4] https://www.foxnews.com (last accessed: 08 July 2022)
[5] https://www.politico.com (last accessed: 08 July 2022)
[6] https://en.news-front.info (last accessed: 08 July 2022)
[7] https://en.newizv.ru (last accessed: 08 July 2022)
[8] https://www.rt.com (last accessed: 08 July 2022)
[9] https://sputniknews.com (last accessed: 08 July 2022)
[10] https://tass.com (last accessed: 08 July 2022)

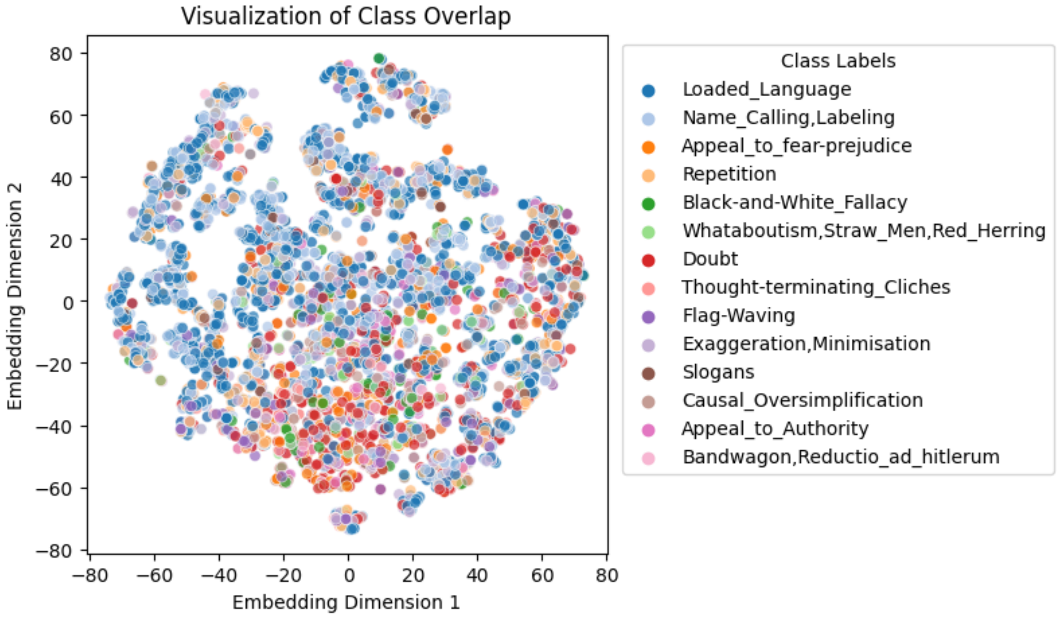


Figure 4.3: Visualization of the Propaganda Techniques' Overlap

4.3.1 Propaganda Technique Corpus

The first step to preparing the date for model training is to address any identified quality issues during the Data Understanding stage.

Data Cleaning

The Propaganda Technique Corpus appears free of issues concerning missing values, duplicate articles, or data entry errors.

Da San Martino et al. [DSMYBC$^+$19] suggest that eliminating negative samples in their experiments improves outcomes. Outlier Detection for specific propaganda classes could help remove propaganda spans that generate excessive noise. Visualizing the overlap of the 14 propaganda techniques reveals that the classes' Embeddings significantly overlap, making it hard to identify all 14 clusters. The extensive overlap of propaganda techniques can be observed in Figure 4.3. Because professional annotators labeled the Propaganda Technique Corpus, it is reasonable to assume that the various propaganda classes are annotated accurately. Attempting to remove overlapping spans results in losing approximately 5,000 training examples. While the data may appear cleaner, Minority Classes are lost, leaving only the Majority Classes left. Consequently, overlapping spans are not removed.

Data Integration

Since the Propaganda Technique Corpus is the only source used for training, there is no need to align the schema of different sources. Also, Data Concatenation and Entity Resolution are not relevant here.

Data Transformation

Transforming the data is necessary to ensure it is suitable for analysis and modeling. In the specific case of using Language-based Transformer Models like RoBERTa, the spans, which are categorical variables, must be encoded to be usable for the model. Normalization, which is the process of scaling numeric variables, is unnecessary since no numeric variables exist in the dataset. For the same reason, Date and Time Conversion is not needed.

Feature Engineering

Feature Engineering involves the creation of new variables or features that may improve the performance of the Data Mining models. In the case of the Propaganda Technique Corpus, a new context feature was introduced, which takes in the tokens left and right of the propaganda span until the whole sentence carrying the propaganda span is found.

The second crafted feature is the length of a propaganda span. As seen in Figure 4.2, the propaganda techniques tend to have different lengths, which could be a helpful indicator for the Propaganda Detection Model.

4.3.2 Analysis Dataset

The Analysis Dataset, a collection of scraped articles from the web, needs more effort to process. The structure and schema of the Propaganda Technique Corpus were used as a guideline to create a suitable dataset to be used for Inference with the Propaganda Detection Model.

Data Cleaning

After scraping the Analysis Dataset, the content is split into the title, an optional subtitle, and the article content. Further, a publishing date, a unique identifier, and an article link were saved during scraping. During scraping, duplicates were ignored, and during Data Cleaning, this was verified. Finally, the Analysis Dataset was checked for non-existing values, but all values were set except for the optional subtitle.

A series of customized functions were introduced to improve text data processing. The method includes the implementation of various text cleaning functions, such as adding whitespaces after periods, removing URLs and emojis, replacing trailing whitespaces, replacing newline and tab characters with spaces, removing non-breaking spaces, and adjusting various types of quotation marks to a uniform style.

The articles are split into sentences, where each line represents one sentence. Furthermore, each sentence is tokenized, ensuring that the tokenized sentences do not exceed a maximum token limit of 256. If a sentence exceeds this limit, it is skipped. Otherwise, the model fails to identify the propaganda spans.

Data Integration

During Data Integration, the articles' titles, subtitles, and contents are concatenated and saved into text files. The unique article id is used as the file's name to identify the articles.

Data Transformation and Feature Engineering

Since just passing whole sentences to identify the propaganda techniques in the Analysis Dataset is impossible, it is necessary to identify the propagandistic spans. This limitation was due to the Propaganda Detection Model being trained on propagandistic spans rather than whole sentences. The Span Identification model developed by Chernyavskiy et al. [CIN20] is used to identify the beginning and ending tokens of the propaganda spans.

The authors treat this task as a Sequence Labeling Problem and use a Begin, Inside, Outside (BIO) tagging format. The RoBERTa model is fine-tuned to predict BIO tags for each token in a sentence. However, since the RoBERTa model does not account for the dependency between predicted labels, a CRF is added as an extra layer. This extra layer helps model the relationship between individual token labels and improves predictions. The RoBERTa-CRF model is trained end-to-end, with the CRF receiving logits for each token and predicting the entire input sequence while considering label dependencies. The CRF works with words, so only tokens that start a word are passed to it, while word continuation tokens are skipped [CIN20].

For Data Transformation and Feature Engineering, the propagandistic spans were identified, BIO tags were generated, and the articles were encoded to fit the numerical input needs of the Transformer models.

4.4 Model

The Propaganda Detection Model was built iterative, and different approaches were tested and evaluated. Whenever a rise in Micro F1-Score is detected, the resulting model is used as the new reference model.

4.4.1 The Baseline Model

In the initial stage of this research, a rudimentary model was developed to establish a foundational benchmark for evaluation. This elementary system possesses the following characteristics:

1. Utilization of propaganda span exclusively as input for the Transformer model.
2. Implementation of RoBERTa as the preliminary Language Model, given its efficacy and superior results during the SemEval Challenge [DSMBCW+20].
3. Adopting the RoBERTa-base model, as the training process for RoBERTa-large is financially expensive during the beginning stages of research.

Training Preparation

Building a Deep Learning system based on Transformer-based Language Models first requires downloading the model and its corresponding Tokenizer and setting up the work environment for Hardware Acceleration, if available.

The training dataset was already prepared during Data Preprocessing, and to evaluate the *Baseline Model*, only the propaganda span was used as model input.

The input is tokenized since Transformer-based Language Models cannot consume characters and strings. The Tokenization process added special tokens and set a maximum allowed length for the input spans. The maximum length parameter of RoBERTa's Tokenizer refers to the maximum number of subwords generated from the input text. Therefore, it is based on the number of subwords, not characters or words. The *Baseline Model* allows for a sequence length of 512 subwords, and the sequence is padded to the maximum length if the input is shorter than the maximum length.

The AdamW optimizer[11] has a Learning Rate of 1e-5 and an Epsilon Rate of 1e-8. Further, a Linear Scheduler without a warm-up is used.

Model Setup

The *Baseline Model* is implemented with PyTorch [PGM+19] and uses a standard training setup. A data loader is used to enumerate through all the batched inputs, and every batch is then passed to the RoBERTa model. The model returns a loss value, calculating an average training loss after all batches are consumed. After every training epoch, a simple evaluation process is executed to evaluate the performance of the current epoch.

Results

The Micro F1-Score is at 0.54 for the *Baseline Model*, but looking at the distinct classes reveals more information. Majority Classes like *Loaded Language*, *Flag-Waving*, *Name Calling/Labeling*, are already predicted relatively well. Even though *Repetition* is within the three most extensive classes, the model struggles to predict the propaganda technique. Also, the model cannot predict four classes at all, namely *Appeal to Authority*, *Bandwagon/Reductio ad hitlerum*, *Black-and-White Fallacy* and *Whataboutism/Straw*

[11] https://pytorch.org/docs/stable/generated/torch.optim.AdamW.html (last accessed: 29 May 2023)

Table 4.1: Classification Report Baseline Model

Class	Precision	Recall	F1-Score	Support
Appeal to Authority	0.0	0.0	0.0	14
Appeal to fear-prejudice	0.32	0.41	0.36	44
Bandwagon/Reductio ad hitlerum	0.0	0.0	0.0	5
Black-and-White Fallacy	0.0	0.0	0.0	22
Causal Oversimplification	0.29	0.28	0.29	18
Doubt	0.42	0.67	0.51	66
Exaggeration/Minimisation	0.52	0.51	0.52	68
Flag-Waving	0.68	0.68	0.68	87
Loaded Language	0.66	0.85	0.75	325
Name Calling/Labeling	0.63	0.76	0.69	183
Repetition	0.38	0.2	0.26	145
Slogans	0.67	0.2	0.31	40
Thought-terminating Cliches	0.25	0.06	0.1	17
Whataboutism/Straw Men/Red Herring	0.0	0.0	0.0	29
Micro F1-Score	0.53	0.58	0.54	1063

Men/Red Herring. All other classes have scores below 0.5. Table 4.1 shows an overview of the results.

This setup is used as a starting point. In the following chapters, only the changes regarding the setup of the *Baseline Model* are described.

4.4.2 Testing Hyperparameter Optimization

This section describes Hyperparameter Tuning for the Learning and Dropout Rate.

Optimizing the Learning Rate

Hyperparameter Tuning is performed on the *Baseline Model* using the Hyperopt library[12]. The first Hyperparameter used for tuning is the Learning Rate. Using a Logarithmic Distribution to find the best Learning Rate for the model, a new Learning Rate is tested with a fresh model, optimizer, and scheduler. The model is trained for a predefined number of epochs. The final training loss and the Micro F1-Score are returned.

Model Changes

The Learning Rate for the *Baseline Model* is changed to 2.4e-05.

[12] http://hyperopt.github.io/hyperopt/ (last accessed: 13 April 2023)

Table 4.2: Classification Report Hyperparameter Model

Class	Precision	Recall	F1-Score	Support
Appeal to Authority	0.09	0.07	0.08	14
Appeal to fear-prejudice	0.26	0.18	0.21	44
Bandwagon/Reductio ad hitlerum	0.0	0.0	0.0	5
Black-and-White Fallacy	1.0	0.09	0.17	22
Causal Oversimplification	0.36	0.22	0.28	18
Doubt	0.42	0.67	0.51	66
Exaggeration/Minimisation	0.47	0.6	0.53	68
Flag-Waving	0.75	0.8	0.78	87
Loaded Language	0.76	0.8	0.78	325
Name Calling/Labeling	0.63	0.84	0.72	183
Repetition	0.4	0.23	0.29	145
Slogans	0.62	0.6	0.61	40
Thought-terminating Cliches	0.38	0.18	0.24	17
Whataboutism/Straw Men/Red Herring	0.14	0.03	0.06	29
Micro F1-Score	0.58	0.58	0.58	1063

Results

The *Hyperparameter Model* (see Table 4.2) has a higher Micro F1-Score (0.58) than the *Baseline Model* (0.54) (see Table 4.1), which indicates better overall performance. For some classes like *Black-and-White Fallacy*, *Causal Oversimplification*, *Exaggeration/ Minimisation*, *Flag-Waving*, *Loaded Language*, *Name Calling*, *Labeling*, *Repetition*, *Slogans*, and *Thought-terminating Cliches*, the *Hyperparameter Model* performs better in terms of F1-Score. Notably, the *Black-and-White Fallacy* class saw a significant increase in Precision from 0.0 to 1.0, though its Recall is still low. The *Baseline Model* has a higher F1-Score for the *Appeal to fear-prejudice* class. Both models have low scores for the *Appeal to Authority* and *Whataboutism/Straw Men/Red Herring* classes, but the *Hyperparameter Model* shows minor improvement. The *Doubt* class has the same F1-Score in both models, with no change in Precision and Recall. *Bandwagon/Reductio ad hitlerum* has all zero values for Precision, Recall, and F1-Score in both models, indicating that neither model could correctly classify any instances of this class.

The Learning Rate has an essential influence on the results but is still insufficient to improve the model dramatically.

4.4.3 Optimizing the Dropout Rate

After identifying the optimal Learning Rate for the *Baseline Model*, the Dropout Rate needs to be modified to prevent Overfitting. By default, the Dropout Rate for the RoBERTa-base model is fixed at 0.1. Again the Hyperopt library is incorporated to search for the optimal Dropout Rate. Such a rate must be able to balance learning time and performance gains feasibly. The model is called *Dropout Model*.

Model Changes

The Dropout Rate for the *Baseline Model* is changed from 0.1 to 0.2.

Results

According to Table 4.3, changing the Dropout Rate to 0.2 increases the Micro F1-Score by 0.02 points to 0.60. The most significant change is that no class is zero-predicted after optimizing the Dropout Rate. In contrast, compared with the *Baseline Model*, the model struggled to predict five Minority Classes. The *Dropout Model* performs better in all metrics for *Appeal to Authority*, *Appeal to fear-prejudice*, *Bandwagon/Reductio ad hitlerum*, and *Causal Oversimplification*. The classes *Black-and-White Fallacy*, *Doubt*, *Name Calling/Labeling*, *Repetition*, *Slogans*, and *Whataboutism/Straw Men/Red Herring* show better performance by the *Dropout Model* in F1-Score, but mixed results in Precision and Recall. Both models' equal or nearly equal performance was shown for *Exaggeration/Minimisation*. In contrast, better performance by the *Dropout Model* in Recall and F1-Score was shown by *Flag-Waving* and *Loaded Language*. Finally, only the *Thought-terminating Cliches* class performs better by the *Dropout Model* in Precision and F1-Score but has lower Recall.

Overall, the *Dropout Model* performs better across most classes regarding F1-Score. However, there are cases, such as *Doubt*, *Name Calling/Labeling*, *Repetition*, *Slogans*, and *Whataboutism/Straw Men/Red Herring*, where the *Hyperparameter Model* offers higher Recall, indicating its relative strength in identifying True Positives, but potentially at the expense of more False Positives. Like the Learning Rate, the Dropout Rate influences the results, and optimizing both Hyperparameters helps improve the *Baseline Model*. Combining both Hyperparameters, the *Dropout Model* is part of the Ensemble Strategy.

4.4.4 Testing Context Addition

Now knowing the optimal Learning and Dropout Rate for the *Baseline Model*, the following experiment performed incorporates a context feature with two different setups tested. The first setup adds the context feature directly during the encoding phase to the propaganda span. The model's input is the concatenated propaganda span with its context, split by special tokens. Since this setup performed drastically worse than the *Baseline Model*, it is discarded.

Model Changes

The second setup defines a custom *Context Model* for Sequence Classification. This model utilizes two RoBERTa models and combines their outputs to perform the classification

Table 4.3: Classification Report Dropout Model

Class	Precision	Recall	F1-Score	Support
Appeal to Authority	0.13	0.21	0.16	14
Appeal to fear-prejudice	0.30	0.39	0.34	44
Bandwagon/Reductio ad hitlerum	1.0	0.6	0.75	5
Black-and-White Fallacy	0.22	0.09	0.13	22
Causal Oversimplification	0.4	0.33	0.36	18
Doubt	0.48	0.58	0.52	66
Exaggeration/Minimisation	0.47	0.59	0.52	68
Flag-Waving	0.71	0.83	0.76	87
Loaded Language	0.73	0.81	0.77	325
Name Calling/Labeling	0.65	0.75	0.70	183
Repetition	0.44	0.25	0.32	145
Slogans	0.69	0.50	0.58	40
Thought-terminating Cliches	0.33	0.06	0.10	17
Whataboutism/Straw Men/Red Herring	0.25	0.03	0.06	29
Micro F1-Score			0.60	1063

task. One model takes the propaganda span as input, while the other consumes the context.

The *Context Model* processes the context and span inputs through each RoBERTa model. The outputs of these models are then passed through a Linear Layer to condense the information. Subsequently, the outputs of both models are concatenated and passed through a Dropout Layer for Regularization. A Linear Classifier then processes the concatenated output.

Results

As shown in Table 4.4, the overall Micro F1-Score did not increase. Interestingly, the classes *Appeal to Authority*, *Appeal to fear-prejudice*, *Name Calling/Labeling*, *Loaded Language*, and Flag-Waving. *Thought-terminating Cliches* and *Whataboutism/Straw Men/Red Herring* have unchanged performance. *Bandwagon/Reductio ad hitlerum* is the class that benefits the most, with an F1-Score increase from 0.57 to 0.75.

Unfortunately, the *Context Model* is resource-intensive, making adding further features and functions expensive. Due to running simultaneously two instances of RoBERTa models at the same time, the computational limits of the hardware are reached quickly. Trying to use a larger RoBERTa model or adding more features results in crashes due to out-of-memory exceptions. Considering these limitations, the performance does not increase significantly to justify further optimizing the *Context Model*. Therefore, the optimized *Baseline Model* is used for further experiments. Variations of the *Context Model* are part of the Ensemble Learning strategy.

Table 4.4: Classification Report Context Model

Class	Precision	Recall	F1-Score	Support
Appeal to Authority	0.50	0.21	0.30	14
Appeal to fear-prejudice	0.37	0.43	0.40	44
Bandwagon/Reductio ad hitlerum	1.00	0.40	0.57	5
Black-and-White Fallacy	0.50	0.05	0.08	22
Causal Oversimplification	0.23	0.44	0.30	18
Doubt	0.52	0.48	0.50	66
Exaggeration/Minimisation	0.44	0.54	0.49	68
Flag-Waving	0.68	0.90	0.77	87
Loaded Language	0.68	0.82	0.75	325
Name Calling/Labeling	0.68	0.74	0.71	183
Repetition	0.40	0.23	0.29	145
Slogans	0.65	0.42	0.52	40
Thought-terminating Cliches	0.25	0.06	0.10	17
Whataboutism/Straw Men/Red Herring	0.50	0.03	0.06	29
Micro F1-Score			0.60	1063

4.4.5 Adding Context, Span Length, and Averaged Embedding

After dropping the *Context Model* due to high resource costs, the following model is built on top of the *Baseline Model*. During training, the Classification Token is extracted from the last Hidden Layer, the Averaged Embedding for non-padding tokens is calculated from the remaining Intermediate Layers, and the length of the span is calculated. The features are then concatenated and handed through a Linear Classifier. This model is called *Averaged Embedding Model*.

Model Changes

The most significant change is reintroducing the context feature during Tokenization, the Averaged Embedding for non-padding tokens, using the span length feature, and the concatenation with the Classification Token. Further, Dropout Rates of 0.1 and 0.2 are tested.

Results

The *Averaged Embedding Model* with a Dropout Rate of 0.2 shows a higher Micro F1-Score (0.62) compared to the model with a Dropout Rate of 0.1 (0.61), indicating slightly better overall performance (see Table 4.5 and Table 4.6). This model, with a Dropout of 0.2, shows improved F1-Scores for *Appeal to fear-prejudice*, *Bandwagon/Reductio ad hitlerum*, *Exaggeration/Minimisation*, *Flag-Waving*, *Loaded Language*, *Name Calling/Labeling*, *Repetition*, and *Slogans* classes compared to the *Averaged Embedding Model* with Dropout of 0.1.

Table 4.5: Classification Report Averaged Embedding Model with Dropout of 0.1

Class	Precision	Recall	F1-Score	Support
Appeal to Authority	0.30	0.50	0.38	14
Appeal to fear-prejudice	0.28	0.48	0.36	44
Bandwagon/Reductio ad hitlerum	0.24	0.80	0.36	5
Black-and-White Fallacy	0.25	0.09	0.13	22
Causal Oversimplification	0.41	0.39	0.40	18
Doubt	0.51	0.68	0.58	66
Exaggeration/Minimisation	0.48	0.59	0.53	68
Flag-Waving	0.76	0.78	0.77	87
Loaded Language	0.78	0.72	0.75	325
Name Calling/Labeling	0.69	0.72	0.70	183
Repetition	0.44	0.35	0.39	145
Slogans	0.59	0.55	0.57	40
Thought-terminating Cliches	0.88	0.41	0.56	17
Whataboutism/Straw Men/Red Herring	0.40	0.21	0.27	29
Micro F1-Score			0.61	1063

On the other hand, *Appeal to Authority*, *Black-and-White Fallacy*, *Causal Oversimplification*, *Thought-terminating Cliches*, and *Whataboutism/Straw Men/Red Herring* classes perform worse in the model with a Dropout of 0.2. Especially noteworthy is the *Whataboutism/Straw Men/Red Herring* class, which has all zero values for Precision, Recall, and F1-Score in the Dropout of 0.2 model, indicating that this model could not correctly classify any instances of this class. The *Doubt* class has the same F1-Score in both models, despite changes in Precision and Recall.

While a Dropout Rate of 0.2 worked better for the *Baseline Model*, the *Averaged Embedding Model* generalizes better with a Dropout Rate of 0.1.

Comparing the Hyperparameter optimized *Baseline Model* (*Dropout Model* see Table 4.3) with the better performing *Averaged Embedding Model* with Dropout of 0.1, the following conclusions can be made: With the classes *Appeal to Authority*, *Appeal to fear-prejudice*, *Black-and-White Fallacy*, *Causal Oversimplification*, *Doubt*, *Exaggeration/Minimisation*, *Flag-Waving*, *Repetition*, *Thought-terminating Cliches*, and *Whataboutism/Straw Men/Red Herring* the performance is better with the *Averaged Embedding Model*. In contrast, the remaining classes are better with the *Dropout Model*.

Variations of the *Averaged Embedding Model* are part of the later described Ensembling Strategy.

Table 4.6: Classification Report Averaged Embedding Model with Dropout of 0.2

Class	Precision	Recall	F1-Score	Support
Appeal to Authority	0.23	0.21	0.22	14
Appeal to fear-prejudice	0.32	0.52	0.40	44
Bandwagon/Reductio ad hitlerum	0.50	0.60	0.55	5
Black-and-White Fallacy	0.50	0.05	0.08	22
Causal Oversimplification	0.25	0.33	0.29	18
Doubt	0.55	0.61	0.58	66
Exaggeration/Minimisation	0.47	0.62	0.54	68
Flag-Waving	0.76	0.83	0.79	87
Loaded Language	0.75	0.78	0.76	325
Name Calling/Labeling	0.67	0.75	0.70	183
Repetition	0.50	0.32	0.39	145
Slogans	0.68	0.68	0.68	40
Thought-terminating Cliches	0.75	0.18	0.29	17
Whataboutism/Straw Men/Red Herring	0.00	0.00	0.00	29
Micro F1-Score			0.62	1063

4.4.6 Testing Cost-Sensitive Learning

Even though the model's abilities to classify the Minority Classes increased considerably compared to the *Baseline Model*, the model still struggles to classify at least seven propaganda techniques with a Micro F1-Score greater than 0.5. Cost-Sensitive Learning is introduced to the *Average Embedding Model* to tackle this problem.

Model Changes

For each propaganda technique, a class weight is calculated by counting the occurrences of the distinct techniques in the training dataset. Then the inverse of class frequencies is computed and normalized so that the class weights sum up to 1. Finally, the Cross-Entropy Loss Function is modified to consume the class weights.

Results

Applying Cost-Sensitive Learning has no positive impact on classifying Minority Classes. The model's overall Micro F1-Score is worse, while simultaneously, the model is overfitting quicker.

4.4.7 Testing Undersampling/Oversampling

Since introducing Cost-Sensitive Learning failed to improve the model's ability to predict the Minority Classes, another attempt was made by testing Undersampling the Majority Classes and Oversampling the Minority Classes. This way, equal distribution of the 14

Table 4.7: Classification Report Sampling Model

Class	Precision	Recall	F1-Score	Support
Appeal to Authority	0.30	0.14	0.19	65
Appeal to fear-prejudice	0.32	0.51	0.40	65
Bandwagon/Reductio ad hitlerum	0.67	0.75	0.70	75
Black-and-White Fallacy	0.46	0.22	0.30	72
Causal Oversimplification	0.52	0.27	0.36	81
Doubt	0.44	0.45	0.44	83
Exaggeration/Minimisation	0.37	0.44	0.40	68
Flag-Waving	0.53	0.76	0.62	78
Loaded Language	0.63	0.56	0.59	88
Name Calling/Labeling	0.58	0.71	0.64	78
Repetition	0.48	0.40	0.44	80
Slogans	0.58	0.51	0.54	82
Thought-terminating Cliches	0.27	0.27	0.27	78
Whataboutism/Straw Men/Red Herring	0.37	0.54	0.44	70
Micro F1-Score			0.47	1063

different propaganda classes is achieved. While there are some caveats, like falsifying the real-world situation by skewing the real-world distribution of propaganda technique usage, the approach could improve the Micro F1-Score by giving the Minority Classes a chance to be better identified.

Model Changes

An Imbalanced Dataset Sampler is introduced to the *Average Embedding Model* and the *Dropout Model*, which calculates each propaganda technique's ideal Over- and Undersampling factor. This way, the training data is manipulated to have about 437 training samples per class.

Results

Introducing Undersampling and Oversampling to the Propaganda Technique Corpus yields no further improvements regarding the total Micro F1-Score. While the classes *Appeal to fear-prejudice*, *Bandwagon/Reductio ad hitlerum*, *Black-and-White Fallacy*, and *Whataboutism/Straw Men/Red Herring* perform better with the *Sampling Model*, all other classes are predicted better with the *Averaged Embedding Model* (see Table 4.7 and Table 4.5). The approach is discarded and not used for Ensembling.

4.4.8 Testing Data Augmentation

As previous experiments showed, manipulating the existing data only works up to a specific level. Bringing in new features like the length of the spans and the span's context was previously the most effective way to squeeze out at least some gains in performance.

The next concept tested with the Propaganda Technique Corpus is introducing Data Augmentation as proposed by Daval-Frerot and Weis [DFW20]. During their research, they could not test Backtranslation since back in 2020, and there was no reasonable way to generate those, as they state.

Model Changes

No direct model changes were made. To generate more examples of the Minority Classes, the examples of those were translated into German. In the next step, the examples were translated back into English. This way, examples with minor semantic changes are generated.

Results

While the generation of new training examples led to the best model proposed by Jurkiewicz et al. [JBKG20], generating new training examples with the help of Backtranslation did not help to increase the overall predictive power of the Propaganda Detection Models (see Table 4.8). While the *Augmentation Model* can predict the classes *Appeal to fear-prejudice*, *Bandwagon/Reductio ad hitlerum*, *Black-and-White Fallacy*, *Causal Oversimplification*, and *Loaded Language* better than the *Averaged Embedding Model*, there is no noteworthy increase in performance. Still, due to the variation of input data, an *Augmented Model* is used for Ensembling.

4.4.9 Testing Part-of-Speech Tags

Mapping Part-of-Speech Tags such as nouns, verbs, and adjectives to a continuous vector space can help a model understand the grammatical structure and relationships between words in a sentence, thus improving its performance in predicting propaganda spans.

Model Changes

During the Preprocessing phase, Part-of-Speech Tags are extracted and passed to the model. An additional layer consumes the tags. Then the Part-of-Speech Tags are combined with the Embeddings from the *Dropout Model* and *Averaged Embedding Model*.

Results

Again no performance gains are made (see Table 4.9). Furthermore, most classes decreased performance, and *Thought-terminating Cliches* and *Whataboutism/Straw Men/Red Herring* are not predicted. Only *Bandwagon/Reductio ad hitlerum* benefited from Part-

Table 4.8: Classification Report Augmentation Model

Class	Precision	Recall	F1-Score	Support
Appeal to Authority	0.09	0.14	0.11	14
Appeal to fear-prejudice	0.32	0.52	0.39	44
Bandwagon/Reductio ad hitlerum	0.33	0.80	0.47	5
Black-and-White Fallacy	0.21	0.14	0.17	22
Causal Oversimplification	0.53	0.44	0.48	18
Doubt	0.60	0.47	0.53	66
Exaggeration/Minimisation	0.49	0.56	0.52	68
Flag-Waving	0.78	0.67	0.72	87
Loaded Language	0.72	0.82	0.77	325
Name Calling/Labeling	0.62	0.81	0.70	183
Repetition	0.36	0.14	0.20	145
Slogans	0.54	0.50	0.52	40
Thought-terminating Cliches	0.29	0.12	0.17	17
Whataboutism/Straw Men/Red Herring	0.29	0.17	0.22	29
Micro F1-Score			0.59	1063

of-Speech Tags, making the model irrelevant for further evaluation (see Table 4.5).

4.4.10 Training Additional Models for Ensembling

In the previous steps, different experiments were conducted to find techniques that would be beneficial in improving the Micro F1-Score of the Propaganda Detection Model. Since most failed by making the model worse or did not add significant performance gains, the next possible step is to create multiple weak models and combine them into a single robust predictor. The previously created model setups are trained with different models like BERT, OPT, and RoBERTa with different Hyperparameters and model sizes. To name one example: A combination of the RoBERTa-base model with 125 million parameters and a Hidden Size of 769 could yield performance gains in combination with RoBERTa-large, which has 355 million parameters and a Hidden Size of 1024 [ZWYJ21].

Since generating and combining an indefinite number of models is not feasible, a model qualified for Ensemble Learning must exhibit the following characteristics:

1. Performance of at least 0.57 Micro F1-Score

2. No missed class during evaluation on the validation set

As Table 4.10 shows, nine models with different setups are used for Ensembling Learning. Six models are based on RoBERTa, two make use of OPT, and the last is BERT based.

Table 4.9: Classification Report Part-of-Speech Tags

Class	Precision	Recall	F1-Score	Support
Appeal to Authority	0.06	0.07	0.07	14
Appeal to fear-prejudice	0.32	0.41	0.36	44
Bandwagon/Reductio ad hitlerum	1.00	0.40	0.57	5
Black-and-White Fallacy	0.25	0.05	0.08	22
Causal Oversimplification	0.37	0.39	0.38	18
Doubt	0.42	0.52	0.47	66
Exaggeration/Minimisation	0.39	0.60	0.48	68
Flag-Waving	0.78	0.77	0.77	87
Loaded Language	0.73	0.76	0.75	325
Name Calling/Labeling	0.68	0.73	0.70	183
Repetition	0.40	0.28	0.33	145
Slogans	0.51	0.53	0.52	40
Thought-terminating Cliches	0.00	0.00	0.00	17
Whataboutism/Straw Men/Red Herring	0.00	0.00	0.00	29
Micro F1-Score			0.58	1063

Except for the *Averaged Embedding Models* (Dropout=0.1), the other models' Dropout Rate is fixed at 0.2. For seven models, the Learning Rate is set at 2.4e-5, except for the two *Context Models*, where the same model architecture is used first with a Learning Rate of 3e-5 and second with 2e-5.

4.4.11 Ensembling the Models

For Ensembling Learning, two different strategies are tested: Ensemble Averaging and Meta Classification with Model Stacking. Each model is evaluated on the validation dataset. During Inference, the output right before the final Activation Function, called logits, is extracted. This resulting vector contains the probabilities for the 14 propaganda techniques for every analyzed propaganda span.

Testing Ensemble Averaging

During Ensemble Averaging, the average output from multiple prediction models is calculated by extracting the logits from each model. The Softmax Function is then applied to these averaged logits, transforming them into probabilities ranging between 0 and 1. The class with the highest probability is selected from this resulting prediction vector. This selected class represents the model's prediction for the most likely propaganda technique applicable to the given input.

When comparing the Micro F1-Score of the Ensemble Averaging (Micro F1-Score: 0.63) (see Table 4.11 with the highest Micro F1-Score of the nine models, the *RoBERTa-large Dropout Model* (Micro F1-Score: 0.6171) listed in Table 4.10 the difference is small. Seven

Model Type Architecture	Micro F1-Score	Learning Rate	Dropout
RoBERTa-base Context(1)	59.83%	3e-5	0.2
RoBERTa-base Context(2)	59.36%	2e-5	0.2
RoBERTa-base Averaged Embedding	58.14%	2.4e-5	0.1
RoBERTa-large Averaged Embedding	59.92%	2.4e-5	0.1
RoBERTa-large Dropout	61.71%	2.4e-5	0.2
RoBERTa-large Augmented Dropout	57.01%	2.4e-5	0.2
BERT-large Dropout	61.05%	2.4e-5	0.2
OPT-350M Dropout	61%	2.4e-5	0.2
OPT-1.3B Dropout	59%	2.4e-5	0.2

Table 4.10: Trained Models for Ensembling Learning

out of 14 propaganda techniques are classified with less than 0.5 Micro F1-Score, making it a not trustworthy model.

Testing Meta Classification with Model Stacking

In the second approach evaluated during Ensemble Learning, Meta Classification is employed. Initially, logits from the models are stacked and flattened into a two-dimensional array. Subsequently, Ridge Regression is applied using these flattened logits and the corresponding True Labels. Finally, the propaganda techniques of the input spans are predicted using the trained Ridge Regression Classifier model.

Combining the nine weak models with a Ridge Regression Classifier yields drastic improvements in all classes and an overall Micro F1-Score of 0.78. Also, all Minority classes are predicted with more than 0.50 F1-Score, the worst being *Whataboutism/Straw Men/Red Herring* with 0.57 (see Table 4.12). Ten propaganda techniques have F1-Scores above 0.70. Therefore, the Meta Classification with Model Stacking is a way better Propaganda Prediction Model than the Ensemble Averaging Model (see Table 4.11).

Table 4.11: Classification Report of Ensemble Averaging

Class	Precision	Recall	F1-Score	Support
Appeal to Authority	0.11	0.14	0.12	14
Appeal_to_fear-prejudice	0.39	0.45	0.42	44
Bandwagon/Reductio_ad_hitlerum	0.60	0.60	0.60	5
Black-and-White_Fallacy	0.50	0.09	0.15	22
Causal_Oversimplification	0.35	0.50	0.41	18
Doubt	0.57	0.61	0.59	66
Exaggeration,Minimisation	0.51	0.57	0.54	68
Flag-Waving	0.78	0.84	0.81	87
Loaded_Language	0.72	0.85	0.78	325
Name_Calling,Labeling	0.68	0.76	0.72	183
Repetition	0.46	0.25	0.32	145
Slogans	0.70	0.57	0.63	40
Thought-terminating_Cliches	0.36	0.29	0.32	17
Whataboutism/Straw_Men/Red_Herring	0.25	0.07	0.11	29
Micro F1-Score			0.63	1063

Table 4.12: Classification Report Meta Classification Model

Class	Precision	Recall	F1-Score	Support
Appeal to Authority	1.00	0.71	0.83	14
Appeal to fear-prejudice	0.73	0.55	0.62	44
Bandwagon/Reductio ad hitlerum	1.00	0.60	0.75	5
Black-and-White Fallacy	0.94	0.73	0.82	22
Causal Oversimplification	0.92	0.61	0.73	18
Doubt	0.80	0.68	0.74	66
Exaggeration/Minimisation	0.77	0.74	0.75	68
Flag-Waving	0.85	0.92	0.88	87
Loaded Language	0.79	0.91	0.85	325
Name Calling/Labeling	0.73	0.84	0.78	183
Repetition	0.71	0.56	0.63	145
Slogans	0.77	0.90	0.83	40
Thought-terminating Cliches	1.00	0.53	0.69	17
Whataboutism/Straw Men/Red Herring	0.76	0.45	0.57	29
Micro F1-Score			0.78	1063

Table 4.13: Classification Report Postprocessing Model

Class	Precision	Recall	F1-Score	Support
Appeal to Authority	0.91	0.71	0.80	14
Appeal to fear-prejudice	0.59	0.59	0.59	44
Bandwagon/Reductio ad hitlerum	0.75	0.60	0.67	5
Black-and-White Fallacy	0.94	0.68	0.79	22
Causal Oversimplification	0.80	0.44	0.57	18
Doubt	0.75	0.64	0.69	66
Exaggeration/Minimisation	0.74	0.74	0.74	68
Flag-Waving	0.83	0.86	0.85	87
Loaded Language	0.79	0.90	0.84	325
Name Calling/Labeling	0.75	0.83	0.79	183
Repetition	0.78	0.64	0.70	145
Slogans	0.76	0.80	0.78	40
Thought-terminating Cliches	0.75	0.53	0.62	17
Whataboutism/Straw Men/Red Herring	0.70	0.48	0.57	29
Micro F1-Score			0.77	1063

4.4.12 Adding Postprocessing

For Postprocessing, the solution was adopted by Chernyavskiy et al. [CIN20].

Model Changes

To increase the F1-Scores of *Repetition* and *Slogans*, the entire training dataset is searched for exact matches by removing punctuation, filtering out stopwords, and applying stemming. The Repetition class is assigned if the analyzed span matches at least two other spans. If only one match exists, the label is predicted with a threshold of at least 0.001. Also, if no match is found, the probability is set to zero unless the prediction probability of the Deep Learning model is at least 0.99. The *Slogans* technique is boosted by 0.5 if the span starts with a hashtag.

Results

When comparing the Classification Report of Meta Classification (see Table 4.12) and of Postprocessing (see Table 4.13) only the *Repetition* and *Name Calling/Labeling* techniques are predicted better. In contrast, all other classes are predicted worse, except *Whataboutism/Straw Men/Red Herring*, which has the same F1-Score in both reports. The overall Micro F1-Score drops by 0.01 to 0.77.

For Inference, having better predictions for eleven than two classes is desirable. Therefore Postprocessing will be discarded when analyzing the Analysis Dataset.

4.5 Evaluation

The chosen metrics for evaluation are Precision, Recall, F1-Score, and Support. These metrics are selected to provide a comprehensive view of the model's classification performance, considering each class's positive and negative predictions.

4.5.1 Evaluation of the Propaganda Detection Model

The Classification Report, presented in Table 4.12, shows the Meta Classification Model's performance on each propaganda technique.

The model achieved a Micro F1-Score of 0.78, indicating good performance in classifying the various propaganda techniques.

However, the performance varies across different classes. The model performs well on techniques such as *Loaded Language* (F1-Score: 0.85) and *Flag-Waving* (F1-Score: 0.88), while it struggles with classes like *Whataboutism/Straw Men/Red Herring* (F1-Score: 0.57) and *Repetition* (F1-Score: 0.63). The varying performance suggests that the model can be further improved to better recognize underperforming classes.

Analysis of Precision and Recall per Class

This chapter analyzes the distinct classes based on Precision and Recall. The thresholds set for the different clusters are relative.

High Precision and High Recall: High Precision means the model is good at accurately predicting positive instances, while high Recall means the model is good at capturing most of the positive instances.

- *Flag-Waving* (Precision: 0.85, Recall: 0.92)

Higher Precision but relatively Lower Recall: Again, higher Precision means the model accurately predicts positive instances, while lower Recall means the model is worse at capturing most positive instances.

- *Appeal to Authority* (Precision: 1.00, Recall: 0.71)
- *Bandwagon/Reductio ad hitlerum* (Precision: 1.00, Recall: 0.60)
- *Thought-terminating Cliches* (Precision: 1.00, Recall: 0.53)
- *Causal Oversimplification* (Precision: 0.92, Recall: 0.61)
- *Black-and-White Fallacy* (Precision: 0.94, Recall: 0.73)
- *Doubt* (Precision: 0.80, Recall: 0.68)
- *Exaggeration/Minimisation* (Precision: 0.77, Recall: 0.74)

Relatively Lower Precision but High Recall: Relatively lower Precision means the model is worse at accurately predicting positive instances, while high Recall means the model is good at capturing most positive instances.

- *Loaded Language* (Precision: 0.79, Recall: 0.91)
- *Slogans* (Precision: 0.77, Recall: 0.90)
- *Name Calling/Labeling* (Precision: 0.73, Recall: 0.84)

Relatively Lower Precision and very Low Recall: Relatively lower Precision means the model is worse at accurately predicting positive instances, while very low Recall means the model is terrible at capturing most positive instances.

- *Whataboutism/Straw Men/Red Herring* (Precision: 0.76, Recall: 0.45)
- *Appeal to fear-prejudice* (Precision: 0.73, Recall: 0.55)
- *Repetition* (Precision: 0.71, Recall: 0.56)

Analysis of F1-Scores per Class

Analyzing the F1-Score for each class helps understand how well the model is doing regarding both False Positives and False Negatives for each propaganda technique. The distinction into High, Moderate, and Low is made by applying the following thresholds:

- F1-Score > 0.8: High
- 0.6 <= F1-Score <= 0.8: Moderate
- F1-Score < 0.6: Low

Those thresholds are relative and only apply to the results of the Classification Report of the Meta Classification.

High F1-Score:

- *Flag-Waving*: 0.88
- *Loaded Language*: 0.85
- *Appeal to Authority*: 0.83
- *Slogans*: 0.83
- *Black-and-White Fallacy*: 0.82

These classes indicate that the model performs well in balancing Precision and Recall, leading to a high Micro F1-Score. High F1-Scores suggest that the model can effectively classify these propaganda techniques.

Moderate F1-Score:

- *Name Calling/Labeling*: 0.78
- *Exaggeration/Minimisation*: 0.75
- *Bandwagon/Reductio ad hitlerum*: 0.75
- *Doubt*: 0.74
- *Causal Oversimplification*: 0.73
- *Thought-terminating Cliches*: 0.69
- *Repetition*: 0.63
- *Appeal to fear-prejudice*: 0.62

The moderate F1-Scores for these classes indicate that the model's performance is neither particularly strong nor weak. The model's ability to classify these propaganda techniques is good, but there is room for improvement.

Low F1-Score:

- *Whataboutism/Straw Men/Red Herring*: 0.57

The low F1-Scores for this class show that the model has difficulty effectively classifying the *Whataboutism/Straw Men/Red Herring* technique. Low F1-Scores may arise from the model's inability to balance Precision and Recall, resulting in many False Positives, False Negatives, or both.

Comparison with the Baseline Model

A comparison between the *Baseline Model* Classification Report (see Table4.1) and the *Meta Classification Model's* Classification Report (see Table 4.12) reveals improvements in the Deep Learning system's performance in analyzing propaganda usage in news articles.

The *Meta Classification Model* (see Table 4.12) compared to the *Baseline Model* (see Table 4.1) is better any predicting any of the 14 propaganda techniques, as the following list shows:

- *Appeal to Authority*: The F1-Score increased from 0.0 to 0.83.
- *Appeal to fear-prejudice*: The F1-Score increased from 0.36 to 0.62.

- *Bandwagon/Reductio ad hitlerum*: The F1-Score increased from 0.0 to 0.75.
- *Black-and-White Fallacy*: The F1-Score increased from 0.0 to 0.82.
- *Causal Oversimplification*: The F1-Score increased from 0.29 to 0.73.
- *Doubt*: The F1-Score increased from 0.51 to 0.74.
- *Exaggeration/Minimisation*: The F1-Score increased from 0.52 to 0.75.
- *Flag-Waving*: The F1-Score increased from 0.68 to 0.88.
- *Loaded Language*: The F1-Score increased from 0.75 to 0.85.
- *Name Calling/Labeling*: The F1-Score increased from 0.69 to 0.78.
- *Repetition*: The F1-Score increased from 0.26 to 0.63.
- *Slogans*: The F1-Score increased from 0.31 to 0.83.
- *Thought-terminating Cliches*: The F1-Score increased from 0.1 to 0.69.
- *Whataboutism/Straw Men/Red Herring*: The F1-Score increased from 0.0 to 0.57.

Overall Micro F1-Score: The Micro F1-Score for the *Meta Classification Model* (0.78) indicates a notable improvement over the *Baseline Model* (0.54). This improvement demonstrates that the *Meta Classification Model* is more effective at classifying propaganda techniques in news articles.

4.5.2 Addressing Business Context

The following section resolves the identified Business Contexts in Chapter 4.1.

Business Context 1: Addressing Media Distrust

The first Business Context, which addresses media distrust, is effectively tackled by successfully identifying the various propaganda techniques in news articles.

For example, recognizing *Loaded Language* can facilitate a more objective understanding of information by revealing potential bias. *Exaggeration/Minimisation* can prevent recipients from distorting understanding if adequately recognized. *Appeal to Authority* can encourage the demand for evidence-based information. *Doubt* can be mitigated by organizations and individuals once recognized, working towards reestablishing trust. The misunderstanding of complex issues that arise from *Causal Oversimplification* can be avoided if the technique is identified, thereby promoting a more nuanced approach to complex topics.

Detecting and highlighting propaganda usage is instrumental in enhancing the credibility, transparency, and trustworthiness of media sources. Consequently, this empowers the general public to discern between reliable sources and those disseminating disinformation, fostering a more informed and engaged society.

This achievement is significant for stakeholders such as news organizations, journalists, media regulatory bodies, social media platforms, fact-checking organizations, educational institutions, and the general public. By revealing the underlying propaganda techniques, stakeholders can better comprehend and mitigate the factors contributing to media distrust, ultimately promoting a more transparent and reliable media landscape.

Business Context 2: Preserving Democratic Processes

In addressing the second Business Context, which focuses on preserving democratic processes, identifying propaganda techniques is essential. By successfully detecting and analyzing disinformation and propaganda content, the potential negative impact on democratic processes within the European Union and its Member States can be minimized.

Policymakers and stakeholders can better develop targeted countermeasures by understanding propaganda techniques and strategies. These countermeasures aid in combating disinformation and protect democratic processes, ensuring that the public's decisions are based on accurate information.

The distinct propaganda techniques can harm democratic processes by undermining the public's trust in democratic institutions, spreading disinformation, and fueling polarization: *Loaded Language*, *Name Calling/Labeling* can manipulate public sentiment by connecting strong emotions to specific subjects, parties, or individuals, ultimately leading to misinformed decision-making. *Repetition* and *Exaggeration/Minimization* can contribute to the distortion of facts, making it difficult for citizens to discern between truth and falsehood. *Appeal to fear and prejudice* can incite panic and further widen societal divides. In contrast, *Flag-waving* and *Bandwagon* tactics can exploit national sentiments or population groups to promote biased perspectives. *Causal Oversimplification* and *Thought-terminating Clichés* can stifle critical thinking and meaningful debate, discouraging a comprehensive understanding of complex issues. Techniques such as *Whataboutism/Reductio ad Hitlerum/Straw man* arguments can discredit opponents and obfuscate the truth.

Collectively, these propaganda techniques can potentially corrupt the foundations of democracy by fostering distrust, perpetuating disinformation, and weakening the ability of citizens to make informed decisions based on accurate information. By identifying these techniques, stakeholders such as European Union institutions, national governments, policymakers, political parties, electoral commissions, media organizations, civil society organizations, and the general public can engage in more informed discourse, contributing to a more robust and resilient democratic process.

Business Context 3: Restoring Trust in Institutions

In restoring trust in institutions, identifying and analyzing propaganda techniques is vital. By detecting propaganda in state-published media, awareness of manipulation-free communication can be raised. This increased awareness helps facilitate a more transparent and accountable media environment, ultimately restoring citizens' confidence in institutions.

Understanding the various propaganda techniques empowers stakeholders to recognize and address manipulation in media coverage, such as the attack on Ukraine. For instance, detecting *Loaded Language*, *Name-calling/Labeling* can help uncover attempts to manipulate public sentiment by connecting strong emotions to particular subjects or entities. Identifying *Repetition* and *Exaggeration/Minimization* can reveal efforts to distort facts and mislead the public. Recognizing *Appeal to fear or prejudice*, *Flag-waving*, and *Bandwagon* tactics can expose the exploitation of national sentiments or population groups to promote biased perspectives.

Moreover, stakeholders can encourage a more informed and meaningful discourse by discerning *Causal Oversimplification*, *Thought-terminating Clichés*, and other techniques that stifle critical thinking. This discourse contributes to a more robust and functional democratic system where citizens can make decisions based on accurate information. Consequently, trust in institutions can be restored as the public becomes more aware of manipulation-free communication and transparent media practices.

Business Context 4: Supporting Media Literacy

In conclusion, the ability to identify propaganda techniques through the Deep Learning system developed in this thesis significantly contributes to supporting media literacy. By providing a tool that detects these techniques in online news articles, stakeholders such as media literacy organizations, educational institutions, teachers, students, researchers, media watchdog groups, social media platforms, non-governmental organizations promoting media literacy, and the general public can enhance their understanding and critical evaluation of the information they consume. This, in turn, fosters a more discerning public better equipped to navigate the complex media landscape.

As citizens become more informed and develop the skills to recognize and identify propaganda techniques, they are less susceptible to manipulation and disinformation. Consequently, the media environment becomes more transparent, accountable, and trustworthy, promoting a well-informed society crucial for the functioning of a democratic system. Therefore, the Deep Learning system presented in this thesis is essential for empowering individuals and organizations to pursue media literacy and a more reliable and honest media landscape.

4.6 Deployment

The code used for building the Propaganda Detection Model was uploaded to GitHub[13].

[13] https://github.com/vitalijhein/propaganda-detection-thesis (last accessed: 29 May 2023)

CHAPTER 5

Results

This chapter aims to present the results of the three research questions.

5.1 In which way can the findings of the SemEval 2020 Task 11 [DSMBCW+20] be combined, and to which degree in terms of F1-Score will this combination be able to compete with the results of the Technique Classification subtask?

As described in Chapter 4.5, the final Propaganda Detection Model performs with an 0.78 Micro F1-Score, after combining various approaches encountered in the challenge. Since the evaluation of test data was closed by Da San Martino et al. [DSMBCW+20] after the challenge, scoring the model on test data is impossible.

Therefore, answering the Research Question is only possible on the scores of the development dataset the participants achieved. Table 5.1 lists the best five submissions, which submitted a solution on the development and test dataset. One participant only submitted a solution on the development dataset and did not submit a paper. Therefore, the submission will not be considered, even though it also ranked with 0.6717 Micro F1-Score [DSMBCW+20] . The Propaganda Detection Model scores with a Micro F1-Score of around 0.78. Comparing it to the evaluation scores of the best five submissions shows superior performance in Micro F1-Score. The developed Propaganda Detection Model has no zero-predicted classes. Consequently, the combination of different techniques used by the participants does lead to even better results in terms of Micro F1-Score. In Chapter 6, the results are discussed in greater detail.

Participants	Micro F1-Score	Number of Zero-Classes
Jurkiewicz et al.	0.7046	0
Chernyavskiy et al.	0.6811	0
Li and Xiao	0.6783	0
Raj et al.	0.6717	0
Blaschke et al.	0.6689	1

Table 5.1: Top 5 Participants Development Dataset [DSMBCW+20]

5.2 What are the similarities and differences in using the different propaganda techniques when looking at Russian and American news articles?

5.2.1 Overall Analysis of the News Articles

Table 5.2 presents data on the number of articles, total sentences, detected propaganda, and detected percentage for various news publishers. The first publishers include ABC News, CBS News, CNN International, Fox News, and Politico. In contrast, the second group comprises News Front, Novye Izvestia, Russia Today, Sputnik News, and Tass News.

In summary, two distinct groups of publishers were analyzed. The first group, ABC News, CBS News, CNN News, Fox News, and Politico, had 12956 articles scraped, containing 567251 sentences. Among these, 114121 instances of propaganda were detected, making up an average of 20.11% of the sentences. With 2986 articles and 142043 sentences analyzed, CNN News showed the highest number of detected propaganda, 31041 (Detected Percentage: 21.85%). In contrast, despite having a larger volume of sentences with a total of 193472, ABC News showed fewer detected propaganda elements, with only 33362 (Detected Percentage: 17.24%).

The second group of publishers included News Front, Novye Izvestia, Russia Today, Sputnik News, and Tass News, from which 38457 articles were scraped, yielding 744649 sentences. In this group, 166061 instances of propaganda were detected, constituting 22.30% of the sentences on average. Russia Today, having the largest volume with 395496 total sentences analyzed, demonstrated the highest number of detected propaganda, 91005 (Detected Percentage: 23.01%). On the other hand, Tass News, with 111819 total sentences, showed the least detected propaganda instances, with only 21244 (Detected Percentage: 19%).

An in-depth discussion of the results can be found in Chapter 6. A small Propaganda Detection experiment on articles in Russian language reveals a 27.5% Detected propaganda rate in Russian domestic reporting and 13.12 propaganda techniques per analyzed articles.

Table 5.2: Overall Analysis of the Articles

Publisher	**Scraped Articles**	**Total Sentences**	**Detected Propaganda**	**Detected Percentage**	**Mean Sentences**	**Propaganda per Article**
ABC News	3998	193472	33362	17.24	48.39	8.34
CBS News	772	26346	5469	20.76	34.13	7.08
CNN News	2986	142043	31041	21.85	47.57	10.4
Fox News	4263	157643	34111	21.64	36.98	8.0
Politico	937	47747	10138	21.23	50.96	10.82
Summary	**12956**	**567251**	**114121**	**20.11%**	**43.78**	**8.81**
News Front	1432	30550	6145	20.11	21.33	4.29
Novye Izvestia	1378	47991	9950	20.73	34.83	7.22
Russia Today	20216	395496	91005	23.01	19.56	4.5
Sputnik	7784	158793	37717	23.75	20.4	4.85
Tass News	7647	111819	21244	19.0	14.62	2.78
Summary	**38457**	**744649**	**166061**	**22.30%**	**19.36**	**4.31**

5.2.2 Analysis of the American Articles

Table 5.3 lists the 14 propaganda techniques and their respective frequencies and percentages in the dataset. The technique used most frequently is *Thought-terminating Cliches*, comprising 8.64% (9864 instances) of the total distribution. *Thought-terminating Cliches* involve using phrases to halt argumentation or thinking on a topic.

The next most common technique is *Exaggeration/Minimisation*, representing 7.94% (9062 instances) of the data. This tactic involves overstating or understating elements of a claim or argument to make it appear more or less significant or impactful. *Bandwagon/Reductio ad hitlerum* and *Flag-Waving* are also significantly used, with frequencies of 7.51% (8571 instances) and 7.45% (8507 instances), respectively.

Loaded Language, an appeal to strong emotional implications to sway an audience's perception or opinion, was found in the dataset with 7.11% (8117 instances). *Doubt*, a technique designed to create uncertainty or skepticism, appears with 6.91% (7887 instances).

Appeal to Authority, *Appeal to fear-prejudice*, *Causal Oversimplification*, *Black-and-White Fallacy*, *Repetition*, *Slogans*, *Name Calling/Labeling*, and *Whataboutism/Straw Men/Red Herring* is also common, with *Name Calling/Labeling* appearing only with 6.17%, being the rarest propaganda technique in American news articles.

Table 5.3: Propaganda Techniques and Frequencies Detected in American Articles

Techniques	**Count**	**Percentage**
Appeal to Authority	7893	6.92%
Appeal to fear-prejudice	8271	7.25%
Bandwagon/Reductio ad hitlerum	8571	7.51%
Black-and-White Fallacy	7170	6.28%
Causal Oversimplification	7902	6.92%
Doubt	7887	6.91%
Exaggeration/Minimisation	9062	7.94%
Flag-Waving	8507	7.45%
Loaded Language	8117	7.11%
Name Calling/Labeling	7039	6.17%
Repetition	7496	6.57%
Slogans	8145	7.14%
Thought-terminating Cliches	9864	8.64%
Whataboutism/Straw Men/Red Herring	8197	7.18%

The variation in the usage of propaganda techniques across the five American news publishers - ABC News, CBS News, CNN News, Fox News, and Politico, help develop an understanding of their published propaganda. CBS News employs the technique of *Appeal to Authority* at a much higher rate, while Politico uses it half as much. The *Appeal* *to fear-prejudice* is predominantly used by ABC News, with CNN News implementing it the least. Examining *Bandwagon/Reductio ad hitlerum*, ABC News noticeably takes the lead, with CBS News utilizing it minimally. CBS News, Fox News, and Politico show an inclination toward the *Black-and-White Fallacy*. *Causal Oversimplification* is mainly used by Politico and rarest by CBS News and ABC News. *Doubt* is used by all publishers in a similar amount.

The *Exaggeration/Minimisation* strategy appears favored by CNN News, while Politico uses it less often. In the case of *Flag-Waving*, CNN News exhibits considerable usage, while CBS News shows low application. *Loaded Language* is most frequently found on Fox News, contrasting with CNN News and Politico's lesser usage. Conversely, *Name Calling/Labeling* seems to be employed substantially by Politico, while Fox News uses it the least. *Repetition* is prominently a CBS News technique, whereas ABC News relies less on it. Politico predominantly uses the *Slogans* strategy, while Fox News and CBS News use them less frequently. CBS News and Fox News substantially use *Thought-terminating Cliches*, with Politico at the lower end. Finally, CNN News and Politico utilize *Whataboutism/Straw Men/Red Herring* techniques most frequently, with ABC News showing the least utilization (see Table 5.4).

Table 5.4: Frequency of Propaganda Techniques in American News Outlets

Techniques	**ABC**	**CBS**	**CNN**	**Fox**	**Politico**
Appeal to Authority	6.44%	10.62%	6.97%	7.26%	5.17%
Appeal to fear-prejudice	10.00%	7.39%	5.19%	6.46%	7.06%
Bandwagon/Reductio ad hitlerum	8.98%	5.74%	6.77%	7.01%	7.57%
Black-and-White Fallacy	6.36%	6.98%	4.94%	6.99%	7.37%
Causal Oversimplification	5.92%	5.58%	7.46%	7.22%	8.31%
Doubt	6.74%	6.47%	7.48%	6.42%	7.63%
Exaggeration/Minimisation	7.74%	7.86%	8.98%	7.80%	5.95%
Flag-Waving	6.35%	5.41%	8.84%	7.83%	6.68%
Loaded Language	6.72%	6.91%	6.28%	8.48%	6.46%
Name Calling/Labeling	6.83%	6.60%	7.23%	3.98%	7.87%
Repetition	6.08%	7.52%	6.46%	7.04%	6.43%
Slogans	6.94%	8.10%	7.24%	6.42%	9.37%
Thought-terminating Cliches	9.29%	7.81%	7.65%	9.86%	5.93%
Whataboutism/Straw Men/ Red Herring	5.61%	7.00%	8.52%	7.23%	8.21%

5.2.3 Analysis of the Russian Articles

Table 5.5 outlines the frequency of the detected propaganda techniques found in Russian news articles. The most prevalent technique used is *Loaded Language*, (8.38% or 13776 cases), directly followed by *Appeal to Authority* (8.31% or 13667 cases). The next most frequent tactic is Black-and-White Fallacy (8.01% or 13171 instances). The three rarest detected propaganda techniques are *Repetition* (4.64% or 7622 cases), *Thought-terminating Cliches* (5.68% or 9336 cases), and *Slogans* (6.2% or 10198 cases).

Table 5.5: Propaganda Techniques and Frequencies Detected in Russian Articles

Labels	**Frequency**	**Percentage**
Appeal to Authority	13667	8.31%
Appeal to fear-prejudice	12420	7.56%
Bandwagon, Reductio ad hitlerum	11660	7.09%
Black-and-White Fallacy	13171	8.01%
Causal Oversimplification	11511	7.00%
Doubt	12689	7.72%
Exaggeration/Minimisation	12334	7.50%
Flag-Waving	12089	7.35%
Loaded Language	13776	8.38%
Name Calling/Labeling	11254	6.85%
Repetition	7622	4.64%
Slogans	10198	6.20%
Thought-terminating Cliches	9336	5.68%
Whataboutism, Straw Men, Red Herring	12659	7.70%

Table 5.6: Frequency of Propaganda Techniques in Russian News Outlets

Techniques	**News Front**	**Novye Izvestia**	**Russia Today**	**Sputnik**	**Tass News**
Appeal to Authority	6.88%	8.78%	8.70%	6.16%	7.13%
Appeal to fear-prejudice	4.82%	8.33%	6.46%	9.55%	8.32%
Bandwagon/Reductio ad hitlerum	8.71%	7.46%	5.66%	5.98%	7.06%
Black-and-White Fallacy	7.18%	9.25%	5.73%	7.91%	6.44%
Causal Oversimplification	7.90%	6.54%	7.02%	9.17%	7.31%
Doubt	6.32%	7.58%	8.85%	8.50%	6.07%
Exaggeration/Minimisation	8.97%	7.97%	5.85%	6.43%	7.73%
Flag-Waving	7.69%	7.52%	7.30%	6.49%	5.53%
Loaded Language	9.50%	7.16%	11.02%	7.97%	7.31%
Name Calling/Labeling	5.97%	7.01%	7.47%	5.50%	5.53%
Repetition	5.13%	4.20%	4.14%	7.33%	8.19%
Slogans	6.77%	5.88%	6.63%	5.02%	8.51%
Thought-terminating Cliches	3.92%	5.89%	5.43%	6.66%	8.17%
Whataboutism/Straw Men /Red Herring	10.23%	6.43%	9.72%	7.34%	6.70%

Next, Table 5.6 shows the relative propaganda technique distribution of the five Russian news outlets: News Front, Novye Izvestia, Russia Today, Sputnik, and Tass News.

The *Appeal to Authority* technique is most frequently used at Novye Izvestia and Russia Today, with Sputnik using it the least. *Appeal to fear-prejudice* sees its highest usage at Sputnik and Tass News, contrasting with News Front, which uses it minimally. The *Bandwagon/Reductio ad hitlerum* technique is most frequently used by News Front, with Russia Today using it the least. Novye Izvestia is noted for a dominant use of the *Black-and-White Fallacy*, whereas Russia Today and Tass News appear less inclined toward this technique. The *Causal Oversimplification* strategy is most heavily used by Sputnik, contrasting with Novye Izvestia, which uses it the least. *Doubt* is used significantly more by Russia Today and Sputnik, while Tass News and News Front use it least. *Exaggeration/Minimisation* is favored largely by News Front, with Russia Today implementing it the least.

Regarding *Flag-Waving*, News Front is also the top user, while Tass News uses it the least. The *Loaded Language* technique is most prevalent in Russia Today's content. However, Novye Izvestia utilizes it less frequently. *Name Calling/Labeling* is primarily used by Russia Today, contrasting with Sputnik and Tass News, which show minimal use of this technique. *Repetition* is more prominent in Tass News, while Russia Today and Novye Izvestia use it the least. Further, Tass News uses the *Slogans* technique more frequently, while Sputnik and Novye Izvestia uses it less often. Tass News also leads in using *Thought-terminating Cliches*, with News Front using this technique the least. Finally, *Whataboutism/Straw Men/Red Herring* techniques are often employed by News Front and Russia Today, while Novye Izvestia and Tass News show the least utilization.

5.2.4 Comparison of American and Russian Propaganda Usage

The analysis of propaganda techniques employed in American and Russian articles reveals similarities and differences. The analysis is based on Table 5.7.

The essential propaganda technique in Russian news articles is *Loaded Language*, in contrast to America, where the technique is the eighth detected technique. Looking at the *Appeal to Authority* technique in Russian articles, the technique is the second most used technique, while in American articles, the technique ranks only at place ten. Next, *Black-and-White Fallacy* is the third most used technique in Russian articles, while in American articles, it is one of the least important techniques, the second least detected technique. For *Doubt*, the fourth most used technique in Russian articles, the distribution is similar: In American articles, *Doubt* is only eleventh.

The technique *Whataboutism/Straw Men/Red Herring* is frequently detected by both countries' news publishers, ranking fifth in Russian and sixth in American articles. This is similar to *Appeal to fear-prejudice*, where the technique ranks sixth in Russian and fifth in American articles' propaganda encounters. *Exaggeration/Minimisation* is the seventh most detected technique in Russian articles, and is the most important in American ones. *Flag-Waving* behaves similarly: In Russian articles, eighth, in American ones third.

Nearly identical importance shows the *Causal Oversimplification* technique, tenth in Russian and ninth in American articles. Of less importance in both countries' articles is *Name Calling/Labeling*, being eleventh in Russian and least detected frequency in American articles.

A more considerable difference can be encountered in *Slogans*, rather not important in Russian (twelfth) versus seventh in American news coverage. A considerable difference in importance can be found in *Thought-terminating Cliches*. The technique is the most detected propaganda technique in American articles and the second least in Russian. Finally, *Repetition* being the least detected technique in Russian articles, and the twelfth detected in American ones, the technique does not appear to be a significant factor in both countries.

Table 5.7: Ranking of Russian and American Propaganda Usage

Rank	**Techniques in Russian articles**	**Techniques in American articles**
1	Loaded Language	Thought-terminating Cliches
2	Appeal to Authority	Exaggeration/Minimisation
3	Black-and-White Fallacy	Bandwagon/Reductio ad hitlerum
4	Doubt	Flag-Waving
5	Whataboutism,Straw Men, Red Herring	Appeal to fear-prejudice
6	Appeal to fear-prejudice	Whataboutism,Straw Men, Red Herring
7	Exaggeration/Minimisation	Slogans
8	Flag-Waving	Loaded Language
9	Bandwagon/Reductio ad hitlerum	Causal Oversimplification
10	Causal Oversimplification	Appeal to Authority
11	Name Calling/Labeling	Doubt
12	Slogans	Repetition
13	Thought-terminating Cliches	Black-and-White Fallacy
14	Repetition	Name Calling/Labeling

Analysis-based Conclusion on Russian Propaganda Usage in English-language Articles

The analyzed Russian articles often employ propaganda techniques to influence their audience effectively. Emotionally-charged language manipulates readers' feelings, pushing them toward specific viewpoints. These articles also frequently appeal to authority, validating their claims by citing perceived experts without providing further evidence. Moreover, complex matters are regularly presented as binary choices through the Black-and-White Fallacy, thus oversimplifying the discourse and potentially misleading readers. Techniques like Doubt undermine trust in individuals, institutions, or ideas that contradict the propagated narrative.

The articles also employ strategies like Whataboutism, Straw Men, and Red Herring to divert the focus away from the main issue or introduce irrelevant information to distract the audience. This diversion is often used to dodge criticism or avoid addressing complicated subjects. They also utilize fear and prejudice as tools to sway public sentiment. The articles can bolster support for their preferred narratives by stoking fear and panic about alternatives. Additionally, by exaggerating minor issues and minimizing major ones, they can control public perception and steer their audience's opinion in the desired direction.

Lastly, the analyzed Russian articles often exploit nationalistic sentiments or identities. By aligning their narrative with the readers' national pride or identity, they can rally support for their cause.

Analysis-based Conclusion on American Propaganda Usage

The analyzed American news articles frequently employ Thought-terminating cliches and Slogans to discourage critical thought and suppress meaningful debate. These cliches typically serve to terminate discussions by dissuading disagreement or promoting a simplified understanding. Furthermore, they utilize Exaggeration and Minimization techniques to manipulate the perceived importance of issues. These articles can direct public attention and shape opinion by amplifying some subjects and downplaying others. The Bandwagon and Reductio ad Hitlerum techniques are common; the former encourages readers to adopt mainstream perspectives, while the latter discredits ideas by associating them with opposing groups or concepts.

Flag-Waving is another prominent strategy, exploiting strong national sentiments or appealing to specific population groups to rally readers around a particular cause or idea. This technique often justifies or promotes certain narratives or initiatives.

Similarly to Russian articles, American ones use the Appeal to Fear and Prejudice, invoking fear and panic about potential alternatives and playing on existing biases to garner support for specific narratives.

Techniques like Whataboutism, Straw Men, and Red Herring are deployed to distract from the main argument. These involve accusing the critic of hypocrisy, misrepresenting an opponent's stance to refute the distorted position, or introducing irrelevant details to divert attention from the primary issue.

Slogans are also commonly used, offering short, catchy phrases that encapsulate a specific viewpoint. These Slogans, often laden with labeling and stereotyping, simplify complex issues and appeal to readers' emotions. Lastly, emotionally charged or Loaded Language is frequently used to provoke a potent response from readers, influencing their attitudes and behaviors.

5.3 How did the usage of different propaganda techniques change during the timeline of the Ukrainian War?

Unless stated otherwise, the following precognitions are relevant for any subsequent plots. Firstly, the analysis pertains to articles published from January 2022 through April 2023. The Russian attack on Ukraine commenced on 24 February 2022[1]. The comparisons are made relatively, given that more than three times as many Russian articles were collected. Normalization is done by counting the monthly encountering of propaganda elements and dividing by the yearly count. For 2023, where data is only available until April 2024, the data is projected for the remaining year based on the rate of the first four months.

5.3.1 Aggregated Distribution of Propaganda Technique

Figure 5.1 represents the overall distribution of encountered propaganda techniques in Russian and American news publisher articles. The plot displays the distribution of detected propaganda techniques per month. The left y-axis, which ranges from 0 to 0.5, illustrates this. The right y-axis shows the number of articles per month, highlighting the discrepancy in the volume of articles published by each country. The red line represents Russian articles, while the blue line represents American ones.

The most conspicuous spike in detected propaganda elements is evident in March 2022 from the American news publisher. This is also the only month American publishers produced nearly as many articles as their Russian counterparts throughout the entire time frame.

Before this spike, detected propaganda levels for both countries were comparatively low immediately following Russia's attack on Ukraine. Propaganda began to rise in February 2022, reaching its peak in March 2022. In April 2022, the number of published American articles nearly halved, while Russian articles decreased by a third.

In the following months, the number of published American articles and the detected propaganda techniques steadily declined until August 2022. However, starting in September 2022, the number of American articles and the detected propaganda began increasing again, peaking in February 2023, one year after the attack's onset. Afterward, the detected propaganda in American news articles started declining again.

The distribution of propaganda techniques in Russian articles paints an entirely different picture. While the spike in February 2022 was not as extreme as in the American articles, the usage of propaganda techniques maintained a consistent level. In the following months, the percentage of detected propaganda was generally higher than in the American articles, except for October 2022 and February 2023.

A deeper look at propaganda over time in Russian articles reveals the spike in detected propaganda techniques in February and March 2022. At the beginning of the attack,

[1] https://www.reuters.com/world/europe/events-leading-up-russias-invasion-ukraine-2022-02-28 (last accessed: 05 June 2023)

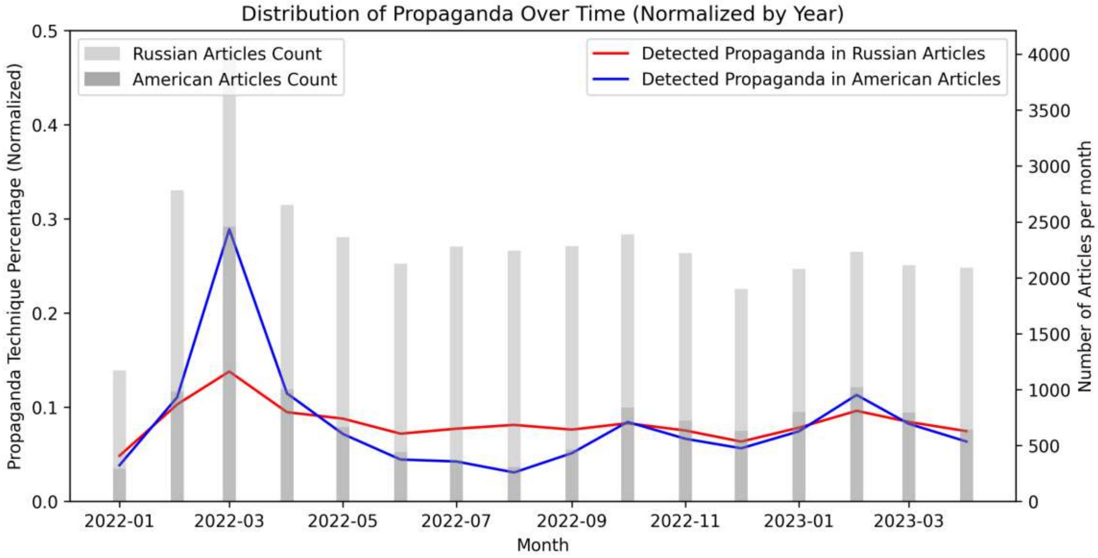


Figure 5.1: Overall Distribution of Propaganda Over Time

Novye Izvestia was the news publisher with the most detected propaganda techniques, followed by Sputnik News, Russia Today, News Front, and Tass. In the following months, detected propaganda decreased drastically in articles by Novye Izvestia, the publisher with the lowest encountered propaganda techniques from August to December 2022. Two spikes in January and March 2023 are returning it to the top propaganda distributor. While Sputnik spikes in February 2022 and 2023, the number of detected propaganda techniques stays stable over the months. Russia Today shows a distribution similar to Sputnik News, with a distinction in July 2022, where the publisher has the most detected propaganda. While News Front is inconspicuous during the attack, the publisher remains among the highest propaganda distributor between May 2022 and February 2023. It is important to note that for News Front, articles only from 24 February 2022 were collected, making the month unreliable. Finally, Tass News shows a relatively low propaganda count detected from January to July 2022. Still, after that, the news publisher quickly grows to one of the biggest propaganda publishers from August 2022 to April 2023. This could be due to restrictions in scraping data before July 2022 (see Figure 5.2).

Figure 5.3 now shows the distribution of detected propaganda per month in American articles. When looking at March 2022, the spike of published articles and detected propaganda indicates a big focus of the news publisher on the attack. The spike of detected propaganda in March 2022 is twice as high as in Russian articles, led by Fox News, Politico, CBS News, CNN News, and ABC News. Interestingly, the articles published by ABC News do not show many propaganda instances until September 2022, when the spike is the highest for them. It also keeps at the higher range for the following months. For American publishers, the attack on Ukraine is not a topic of immense interest, at least when looking at the decreasing number of published articles per month

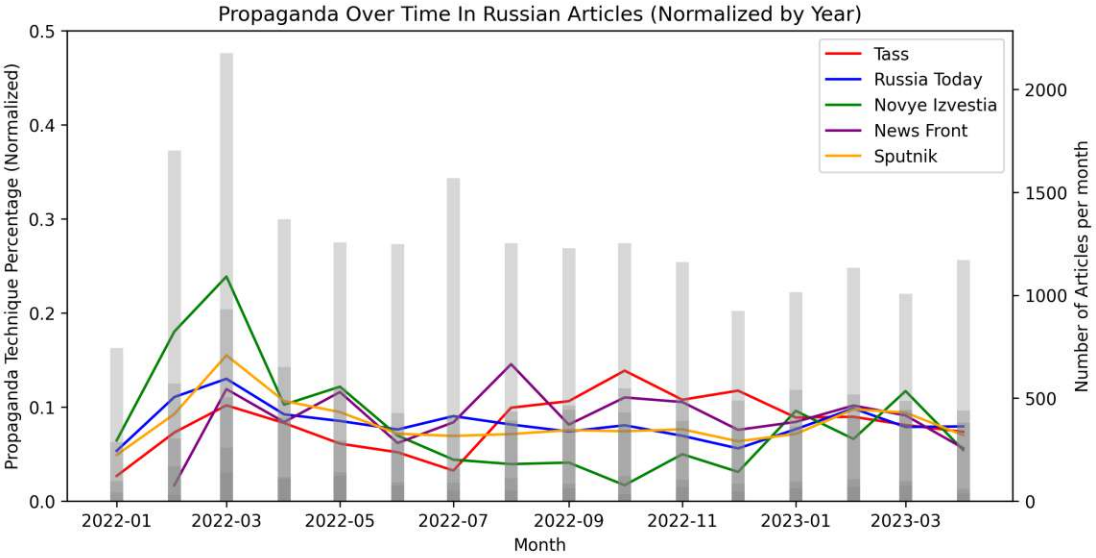


Figure 5.2: Propaganda Over Time in Russian Articles (Normalized by Year)

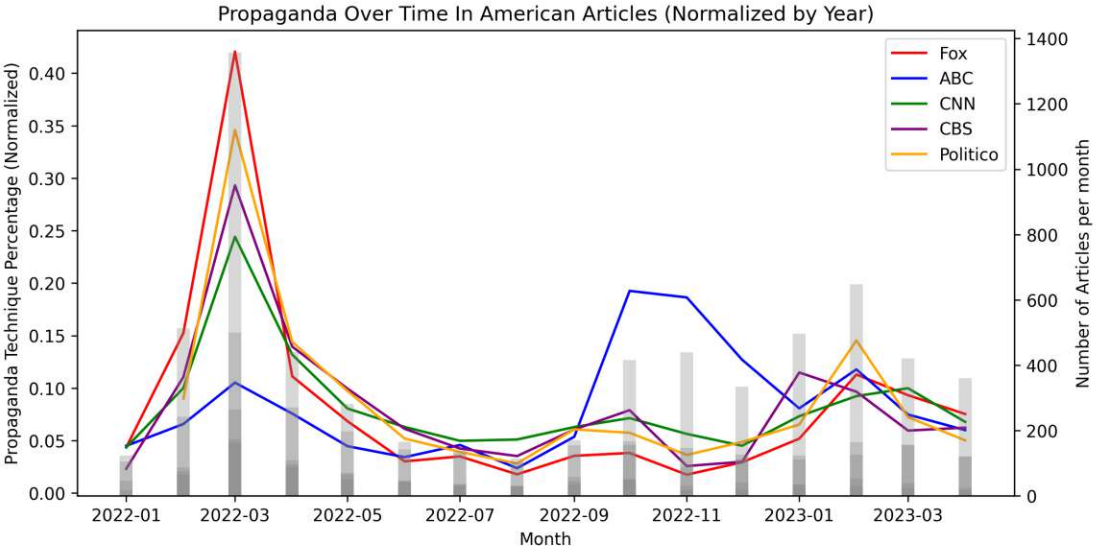


Figure 5.3: Propaganda Over Time in American Articles (Normalized by Year)

from April to August 2022. Then, the total published articles count doubles, while detected propaganda techniques stay low until January 2023, when detected propaganda again grows and spikes in February 2023 for all publishers.

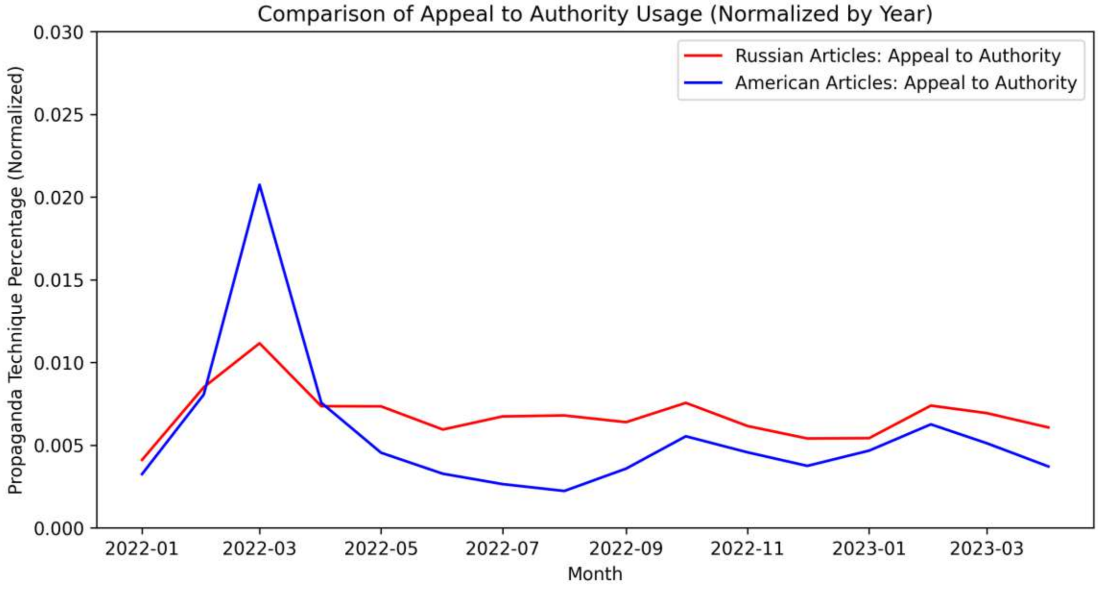


Figure 5.4: Comparison of Appeal to Authority Usage

5.3.2 Monthly Aggregated Distribution per Propaganda Technique

This chapter hosts the plots for the 14 propaganda techniques. Every propaganda technique is analyzed beginning from January 2022 to April 2023. The y-axis ranges from 0 to 0.03 for each plot. The red line always corresponds to the detected propaganda in Russian articles, while the blue line is for American articles. Figure 5.4 shows the frequency of this technique peaks in March 2022 for Russian and American Articles. It then decreases in April 2022 but remains relatively stable until August 2022 for Russian articles. There is a slight rise in October 2022, followed by a decrease towards the end of the year. In 2023, the frequency of this technique shows an increase in February, then a slight decrease through April. The American articles frequency decreases substantially in April 2022 and remains relatively low through the summer. Again there is a slight peak in October 2022, followed by a decrease towards the end of the year. In 2023, there is a rise in the frequency of this technique in January and February, followed by a decrease in March and April.

The distribution of the *Appeal to fear-prejudice* technique can be seen in Figure 5.5. Russian articles show a significant peak in the usage of this technique in March 2022. The frequency then decreases in April 2022 but stays relatively high compared to the start of the year. It fluctuates throughout the rest of the year, with its lowest frequency in June 2022. Starting in 2023, there is a rise in January and February, followed by a decrease in March and April. American articles also show a peak on 22 March. After April 2022, the frequency stays relatively low but has a small peak in October 2022. January and February 2023 show a rise in frequencies, with a subsequent decrease.

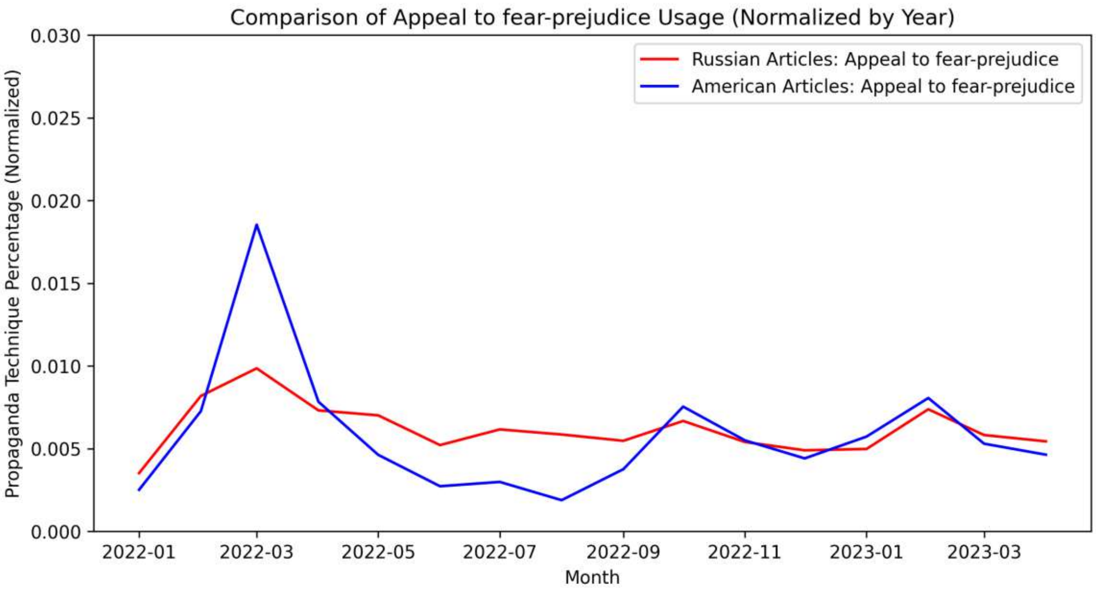


Figure 5.5: Comparison of Appeal to fear-prejudice

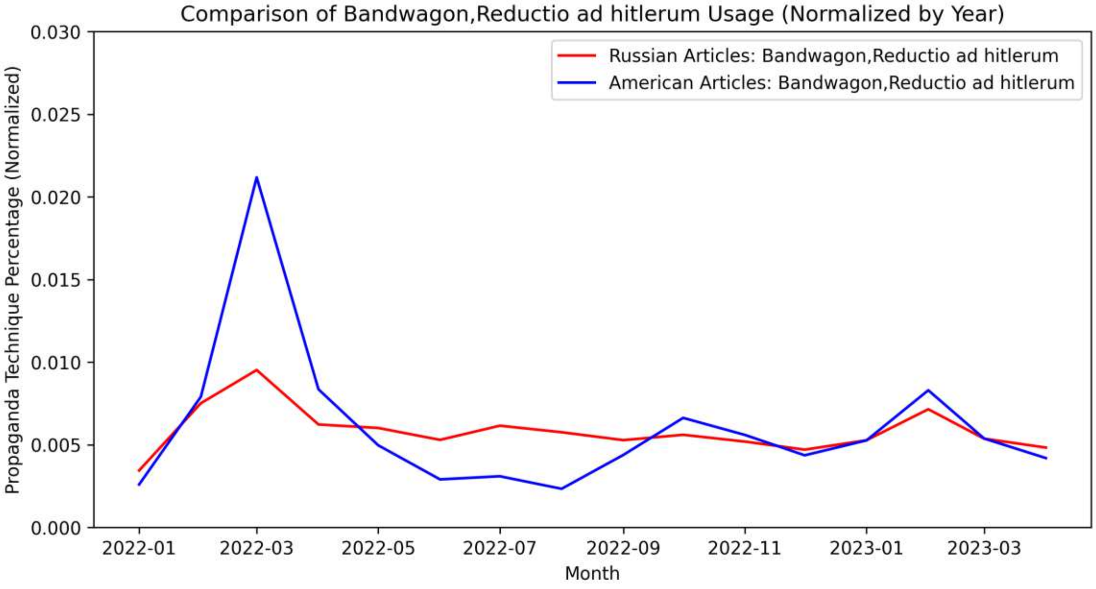


Figure 5.6: Comparison of Bandwagon/Reductio ad hitlerum Usage

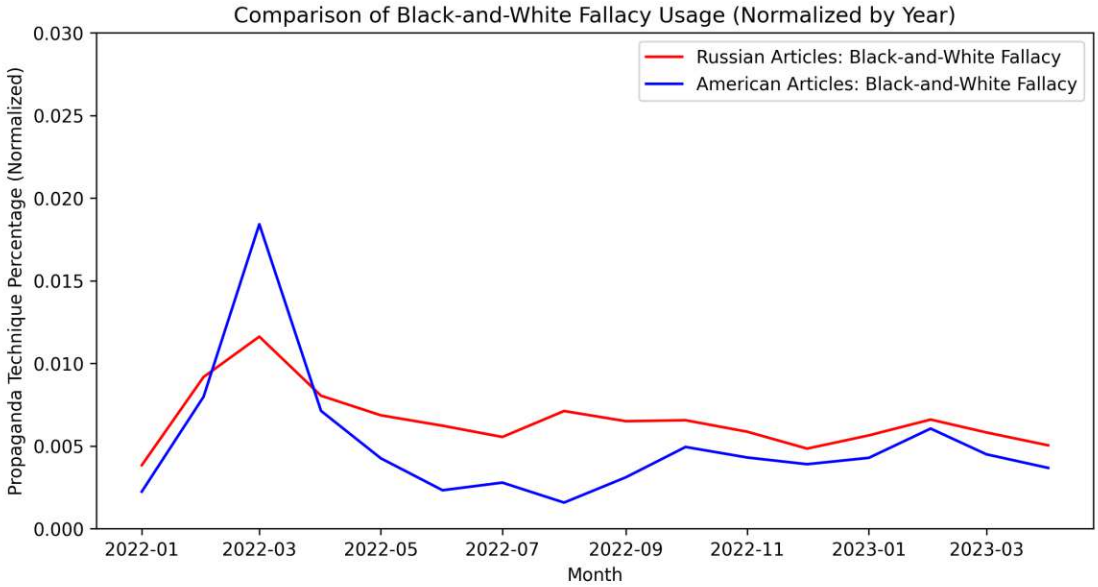


Figure 5.7: Comparison of Black-and-White Fallacy Usage

The propaganda technique *Bandwagon/Reductio ad hitlerum* (see Figure 5.6) was mainly utilized in March 2022, but the peak is significantly higher in American articles. After May 2022, the frequency drops below the Russian articles' frequency, surpassing it only in October 2022 and February 2023. In November 2022, January, and March 2023, both Russian and American articles have a similar technique frequency. The Russian frequency is stable throughout the months, with smaller peaks in July, October 2022, and February 2023.

The technique *Black-and-White Fallacy* (see Figure 5.7) shows an interesting tendency: Except for March 2022, where American technique frequency peaks, Russian article frequency is higher for every remaining month in 2022 and 2023. After March 2022, the frequency of Russian articles decreases but keeps peaking in August, October 2022, and January 2023. American articles' frequency decreases to a minimum in May and August 2022. After August 2022, the technique frequency keeps growing until February 2023. For both countries, the technique frequency decreases in March and April 2023.

Figure 5.8 shows the frequency of the *Causal Oversimplification* propaganda technique. For the Russian articles, the frequency reached its peak in March 2022. Following this peak, there is a decrease in April 2022, but the frequency remains relatively stable, with slight fluctuations through the rest of 2022. From January 2023 to February 2023, there is a slight increase in usage, which then decreases through April 2023. In American articles, the usage of this technique also peaks in March 2022, showing a significantly higher frequency than in Russian articles. The frequency then decreases substantially from April 2022 to August 2022. It peaks again in October 2022 and January 2022.

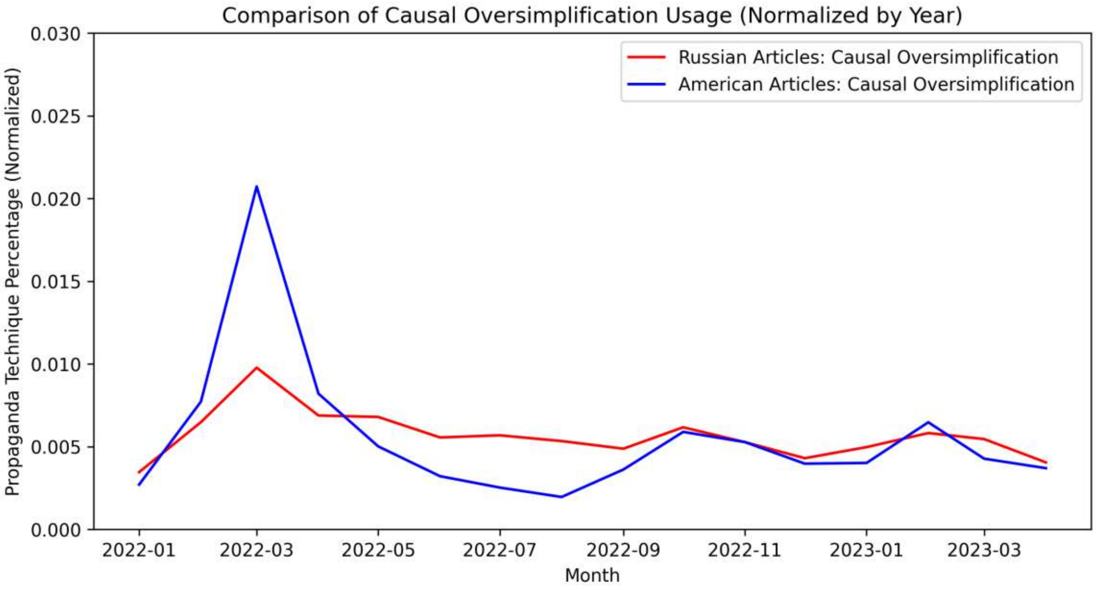


Figure 5.8: Comparison of Causal Oversimplification Usage

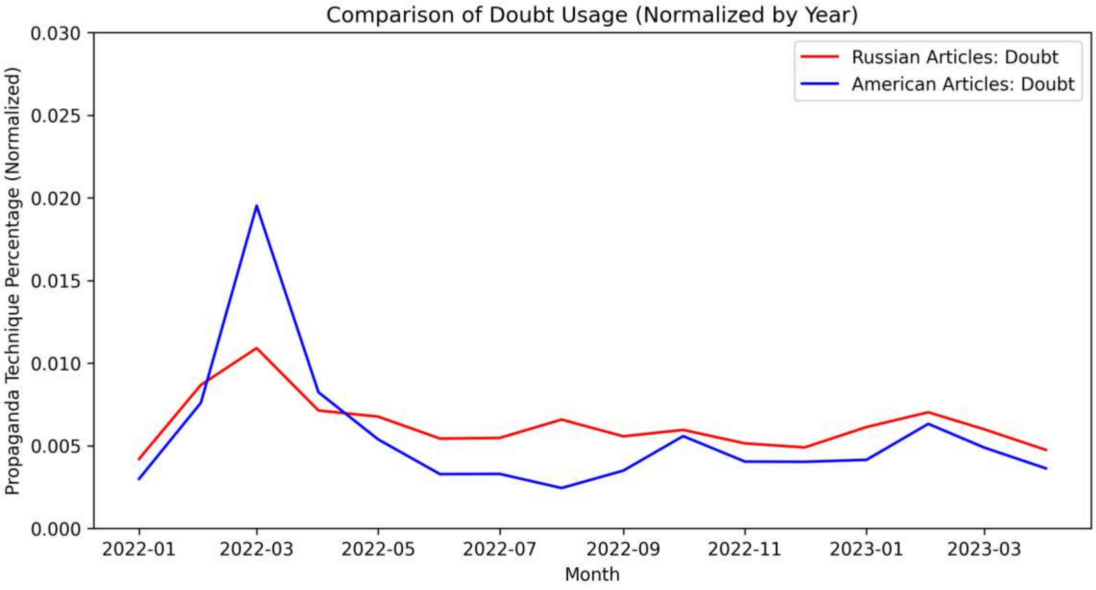


Figure 5.9: Comparison of Doubt Usage

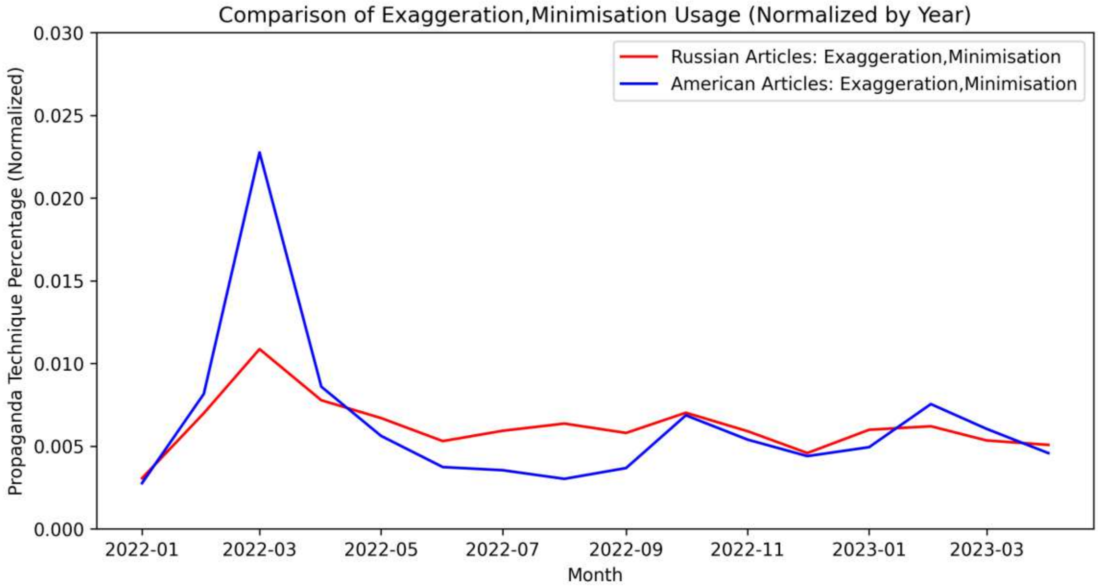


Figure 5.10: Comparison of Exaggeration/Minimisation Usage

Next, the propaganda technique *Doubt* is shown in Figure 5.9. For the Russian articles, there is a significant peak in usage around March 2022, with a frequency slightly above 0.01. The usage frequency decreases afterward, but smaller peaks are observed in August 2022 and February 2023. For the American articles, the usage frequency peaks around March 2022, reaching just under 0.02. This is the highest frequency observed in the plot. There is a significant drop in usage following this peak, and the frequency remains relatively low, with minor peaks in October 2022 and February 2023. After April 2022, American articles always show less usage of *Doubt* than Russian ones.

Figure 5.10 represents the frequency of the *Exaggeration/Minimisation* propaganda technique usage. Again for both countries, there is a peak in March 2022, where the American peak is more than twice higher. The usage decreases from April 2022 until August 2022 and again peaks in October 2022 and February 2023. Russian usage is higher from May 2022 to September 2022, November 2022, January 2022, and January 2023. The remaining months show more usage in American news articles.

Figure 5.11 shows the *Flag-Waving* technique frequency. There is a noticeable peak in usage around March 2022 for Russian and American articles, with a frequency slightly below 0.01 for Russian and above 0.02 for American articles. From May 2022 to August 2022, for American articles, there was a significant drop in usage following this peak. The other peaks are around October 2022 and February 2023. For Russian articles, the usage frequency appears to decrease after the peak in March 2022, with minor fluctuations throughout the rest of 2022 and 2023.

Figure 5.12 shows that Russian and American articles fluctuate using the *Loaded Language*

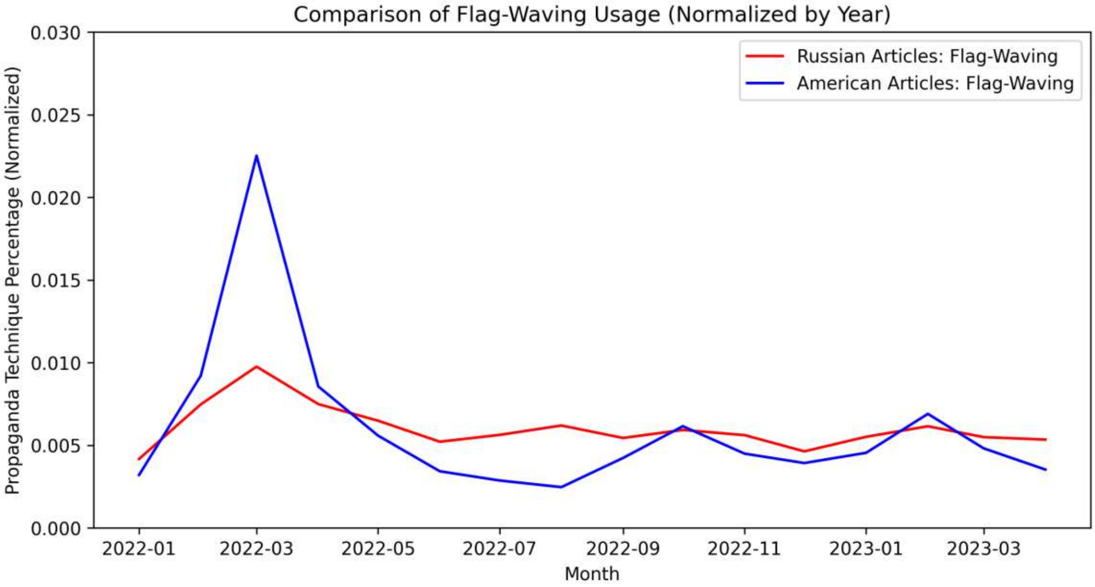


Figure 5.11: Comparison of Flag-Waving Usage

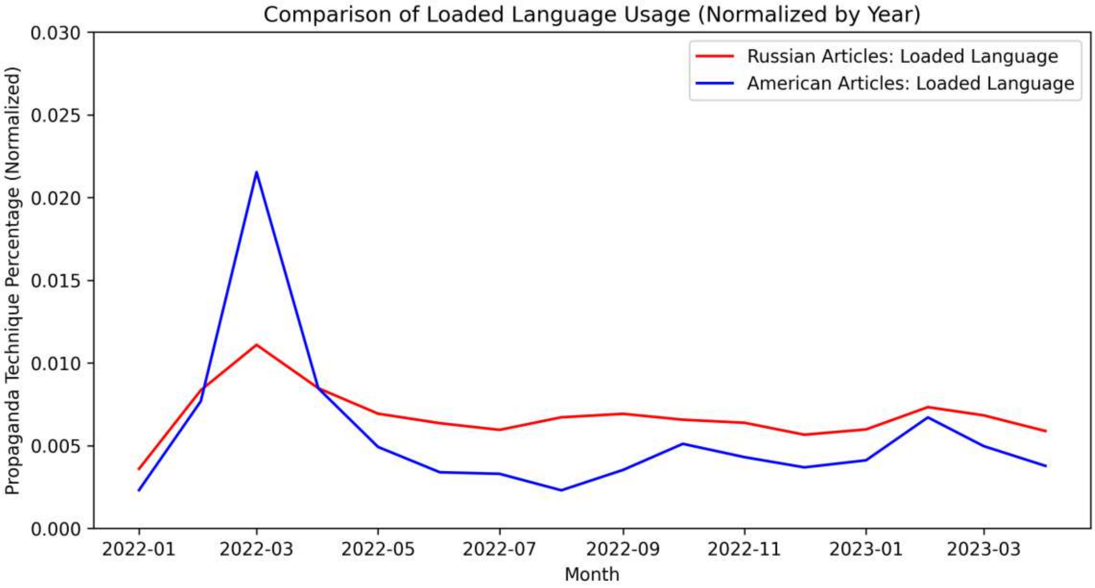


Figure 5.12: Comparison of Loaded Language Usage

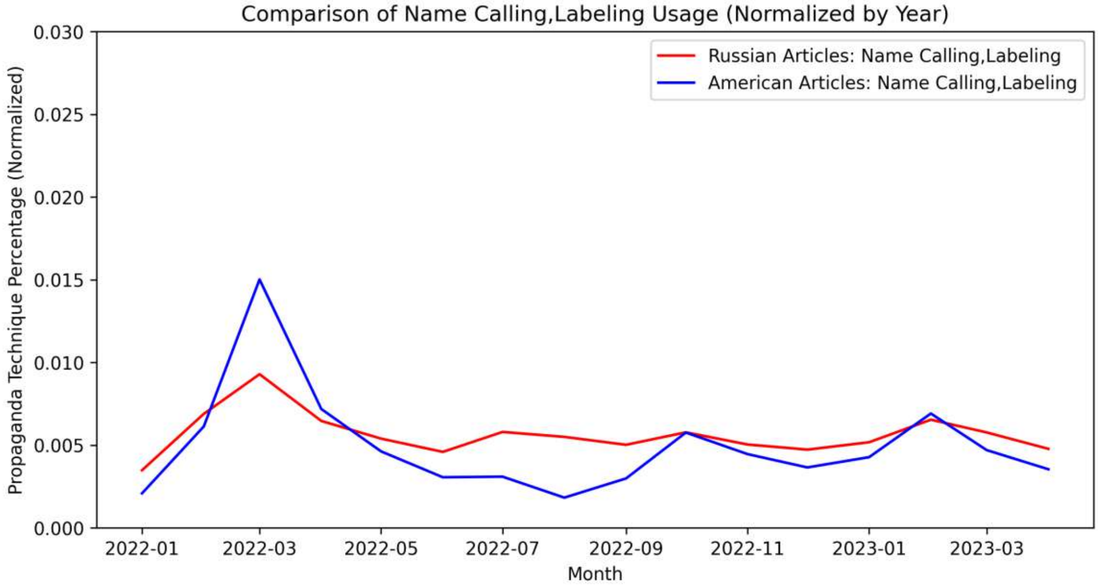


Figure 5.13: Comparison of Name Calling/Labelling Usage

propaganda technique from January 2022 to April 2023. Russian articles show noticeable peaks in usage around March 2022 and February 2023, both slightly above 0.01. The usage frequency generally decreases over the observed period, with minor fluctuations. For the American articles, the usage frequency appears to peak around March 2022, reaching just above 0.02. There is a significant drop in usage following this peak, and the frequency remains relatively stable for the rest of 2022. In October there is a slight increase, followed by a decrease again. In 2023, there is a slight increase in usage in February, followed by a decrease. When comparing the two countries, American articles show a higher peak in using the *Loaded Language* propaganda technique (in March 2022), but their usage frequency decreases significantly afterward. On the other hand, Russian articles exhibit a more consistent usage of this technique over time, with minor fluctuations and a general downward trend.

The *Name Calling/Labeling* technique (see Figure 5.13) shows a significant peak in the usage of this technique in March 2022 for both countries. However, the peak for American articles shows a significantly higher frequency than the Russian articles. The Russian articles' frequency is lower in February 2022, March 2022, and February 2023. In the remaining months, the frequency of usage is higher for Russian articles. American articles show a significant peak in October 2022 and February 2023.

In Russian articles, the *Repetition* technique (see Figure 5.14) initially increases from January 2022, reaching a peak in March 2022. The frequency decreases in April 2022 and continues to fluctuate over the next months, with another small peak in October 2022. There is a substantial drop in December 2022 and January 2023. It increases again in February 2023 and then decreases again until April 2023. American articles show a

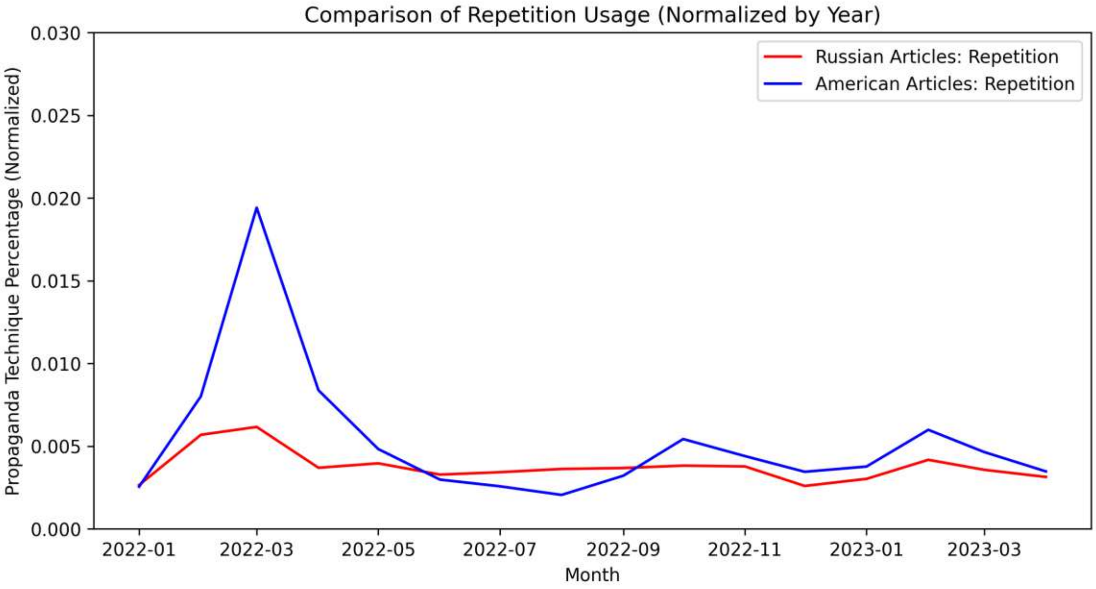


Figure 5.14: Comparison of Repetition Usage

significant peak in the usage of this technique in March 2022, which is more pronounced than the Russian articles. The frequency decreases substantially after April 2022 and keeps decreasing, reaching another peak in October 2022, similar to the Russian trend. From January 2023, there is a rise in usage, followed by a decline through April 2023.

Looking at *Slogans* (see Figure 5.15) again, March 2022 shows the highest peak for both American and Russian articles, but the American frequency is more than twice higher. Russian technique frequency then decrease but stays relatively stable, while American frequency decreases until August 2022, when it reaches its lowest point. In September and October 2022, the usage frequency increases again for both. After that, the frequencies are similar, with slight differences from February to April 2023.

Thought-terminating Cliches being the most used propaganda technique in American articles, Figure 5.16 shows this exactly. American articles utilize the technique more often in all months except from June 2022 to September 2022—the usage frequency peaks significantly around March 2022, October 2022, and February 2023. For the Russian articles, there is a significant peak in usage around March 2022. The usage frequency decreases afterward.

For Russian articles, the *Whataboutism/Straw Men/Red Herring* technique in Figure 5.17 shows a noticeable peak in usage around March 2022. Russian usage of *Whataboutism/Straw Men/Red Herring* appears to be higher in the following months, except for October 2022 and January 202, where American articles show two spikes in usage. In March and April 2022, American publishers also utilized the technique predominantly. Comparing the two countries, American articles show a higher peak

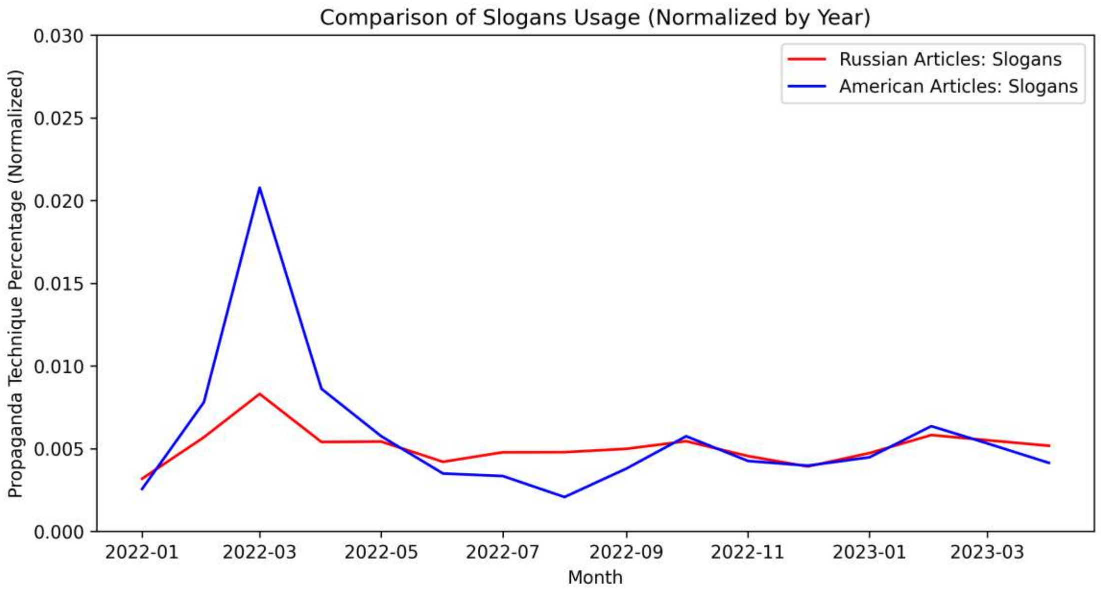


Figure 5.15: Comparison of Slogans Usage

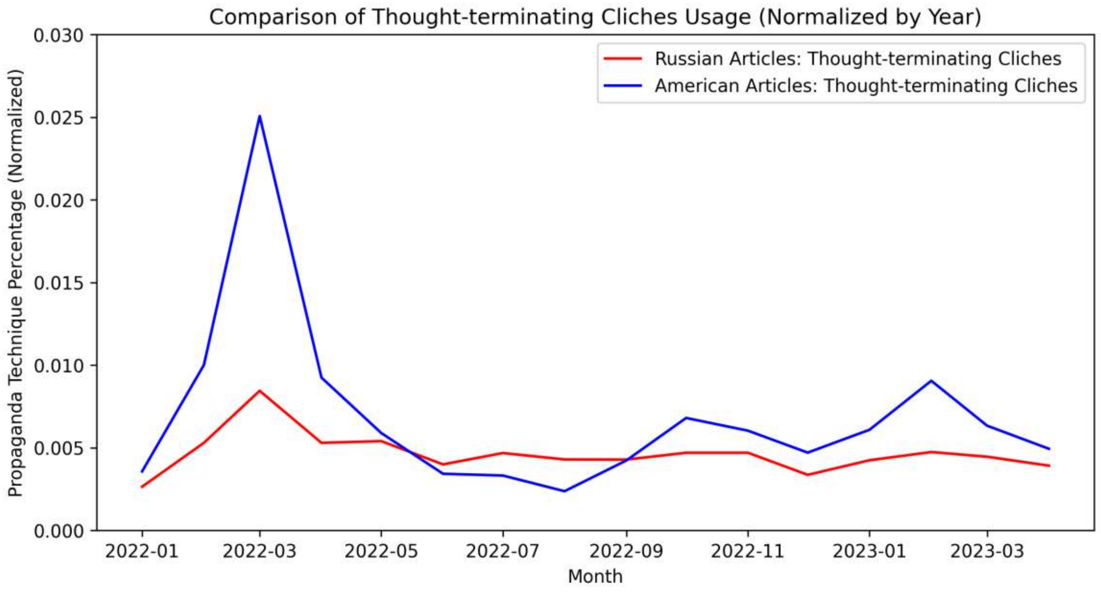


Figure 5.16: Comparison of Thought-terminating Cliches Usage

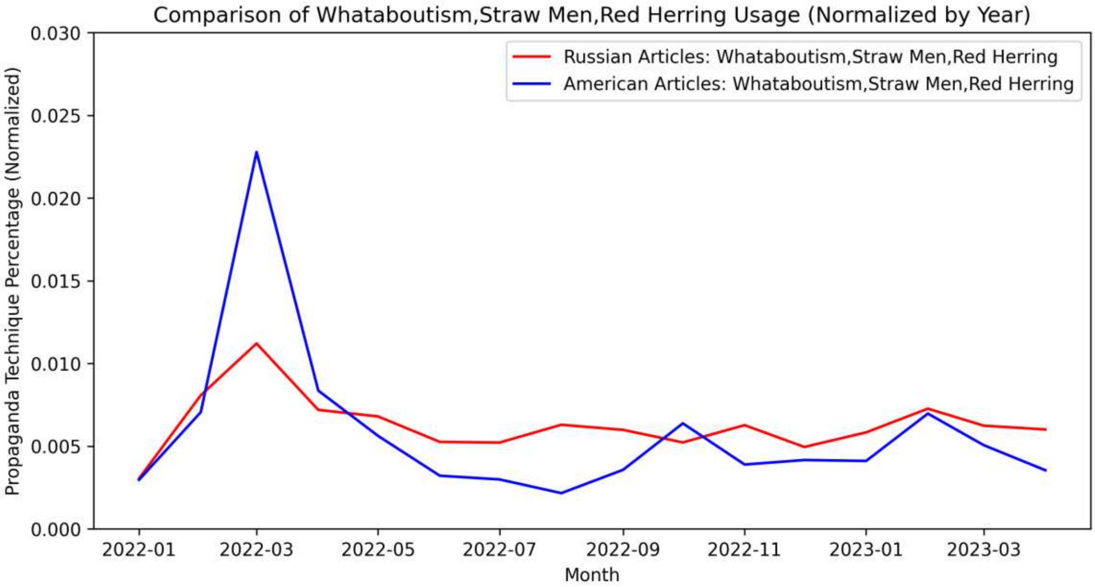


Figure 5.17: Comparison of Whataboutism/Straw Men/Red Herring Usage

in using the *Whataboutism/Straw Men/Red Herring* propaganda techniques (in March 2022), but their usage frequency decreases significantly afterward. On the other hand, Russian articles exhibit a more consistent usage of these techniques over time, with minor fluctuations and a general downward trend.

CHAPTER 6

Discussion

In this chapter, the efforts and issues to construct an optimized model for Propaganda Detection are discussed. Further, limitations and further experiments are thematized.

6.0.1 Combining Deep Learning Approaches

The first Research Question primarily concerns connecting knowledge from different Deep Learning areas to build a superior model.

Most successful participants worked with Transformer-based Language Models, justifying the incorporation of Transfer Learning and Transformer Models. Jurkiewicz et al. [JBKG20], who placed first, and Morio et al. [MMOM20], who placed third, both used a setup with multiple models and Hyperparameter Optimization, thereby forming the starting point for optimizing the Baseline Model. Further, four of the five top submissions used the surrounding context of a propaganda span, indicating its importance in the Propaganda Detection Model.

Next, the Averaged Embedding and Span Length, as described by Chernyavskiy et al. [CIN20], was leveraged, leading to the best-performing model before Ensemble Learning. Evaluating the Literature Analysis, Cost-Sensitive-Learning was referenced by multiple participants like Jurkiewicz et al. [JBKG20], Grigorev and Ivanov [GI20], and Li and Xiao [LX20]. While the participants derived positive results, different attempts to implement it resulted in bad-performing models.

Over- and Undersampling seemed to be reasonable attempts to equalize the difference between Minority and Majority Classes, as proposed by Jiang et al. [JGM20] and Grigorev and Ivanov [GI20]. Furthermore, even though Minority Classes were predicted better by the Propaganda Detection Model, the overall Micro F1-Score decreased substantially.

Upon analyzing the Averaged Embedding Model, substantial improvements were made by adding new features. In their submission, the second-placed Morio et al. [MMOM20] used Part-of-Speech Tags and Named Entity Tags. Creating a Part-of-Speech Embedding and adding it to the Averaged Embedding Model did not help increase the Micro F1-Score. Testing the combination of a Part-of-Speech Embedding with the plain RoBERTa Embedding also did not help.

After previous attempts resulted in no significant improvements, the decision was made to use existing models, such as the optimized Baseline and the Averaged Embedding, and train them using different Transformer-based Language Models. Morio et al. [MMOM20] used this approach to train multiple models and combine them during Ensemble Learning. The existing model architectures were used to train Language Models like BERT and OPT. The resulting weak learners combined during Ensemble Learning led to the superior Propaganda Detection Model.

Lastly, the Postprocessing approach described by Chernyavskiy et al. [CIN20] was tested to improve the predictions for the *Repetition* and *Slogans* classes. Unfortunately, this process improved the performance of the weak learners but not of the Propaganda Detection Model. On the contrary, the Micro F1-Score decreased.

While many strategies were proposed, those implemented by the top five submissions were preferred. Nevertheless, some exciting yet time- and resource-consuming ideas were not implemented.

6.0.2 Dataset and Model Limitations

The following dataset and model limitations apply only to weak learners before using Ensemble Learning. The issues described here are not relevant after that.

The models were trained on articles from American publishers only, possibly resulting in greater robustness with the American writing style. Furthermore, the RoBERTa and BERT models, released in 2019, are not updated with events post-release, potentially reducing their stability in recognizing contemporary contexts. The same problem occurs with the Propaganda Technique Corpus. The dataset was released in 2019 and not updated afterward. Hence, the Propaganda Detection Model was fine-tuned on articles from before 2019, potentially favoring the recognition of Majority Classes over Minority Classes. As news articles emphasize current events, the model might struggle to apply its learning from historical contexts to current ones.

Despite the dataset's robust base, model enhancement requires more labeled data, particularly given the challenges faced by Minority Classes before Ensemble Learning. Due to the highly imbalanced Propaganda Technique Corpus, weak learners trained on this dataset tend to replicate technique distribution within unseen data. For example, the propaganda technique *Loaded Language* made up about 50% of occurrences across all ten analyzed news publishers while also being the most prevalent technique in the Propaganda Technique Corpus. The dominance of this class becomes apparent when considering the span length and some examples, which often consist of single words. Consequently, over-descriptive, loaded adjectives may be employed more frequently to create engaging articles. In contrast, a sentence containing a technique like *Whataboutism* requires a more complex sentence structure, beginning with an argument needed to be relativized, followed by a sentence that can relativize the previous argument.

Regarding the Propaganda Technique Corpus, it is uncertain whether its class distribution accurately reflects the distribution of propaganda techniques in news articles over time or merely captures a snapshot of American news articles published before 2019. Therefore, it is unclear if a more balanced distribution of propaganda techniques could lead to better prediction of Minority Classes at the expense of distorting real-world propaganda usage. Similarly, it is unknown if the current class imbalance should be preserved while increasing the number of examples according to the class distribution. Finally, it is also not said that including more recent news articles would help improve Propaganda Detection Models, even though the suggestion was made previously to enhance predictions of Minority Classes.

Examining the Analysis Dataset reveals potential differences in publishing styles that could further complicate matters. For instance, Tass News publishes shorter statements with more formal language. In contrast, Fox News tended to write lengthy articles with descriptive language and ample room for interpretation and author opinions. The limited availability of English-language Russian news publishers contributed to these discrepancies. Furthermore, the lack of a labeled propaganda dataset in the Russian language hinders the analysis of articles written in Russian.

Lastly, the Span Identification model proposed by Chernyavskiy et al. [CIN20] may pose a challenge, as the model's prediction errors can carry over to the Technique Classification model. This issue arises because potentially incorrect spans served as inputs for classification, increasing the likelihood of erroneous predictions. Since the Propaganda Detection Model was trained on propaganda spans, it struggled to classify techniques accurately by inputting the whole propaganda sentence. Additionally, both models were trained on the same imbalanced dataset, potentially leading to a bias toward Majority Classes during Span Identification.

6.0.3 The Issue with Micro F1-Score in the SemEval Challenge

The Micro F1-Score, though seemingly effective, encounters a crucial issue: its enhancement can be achieved if the model disregards Minority Classes. Consequently, the Micro F1-Score increased due to its consistent prediction of Majority Classes while becoming entirely unreliable for Minority Classes. In the context of the SemEval challenge, participants could potentially attain high Micro F1-Scores by sacrificing Minority Classes. Although this strategy might secure victory within the competition, it raises doubts about its practicality in the broader scope of Propaganda Detection. As a result, for future contests on the Propaganda Technique Corpus, challenge organizers should incorporate an additional criterion: "Highest Micro F1-Score with every class classified".

6.0.4 Computational Power and Resources

Using BERT and RoBERTa, developed in 2018 and 2019, respectively, as the primary models for building the Propaganda Detection Model might raise questions. First, when the development process began, Artificial Intelligence was a specialized topic in public discourse. Public discussions focused on Large Language Models like GPT-3 and their potential future applications. The release of ChatGPT[1], a fine-tuned version of GPT-3 capable of interacting in a chat environment, spurred excitement around AI and strengthened scientific research. Subsequently, GPT-4 was introduced, marking the most successful Transformer Model ever created. Also, institutions like Meta unveiled OPT, a remodeled version of GPT-3, designed to demonstrate model construction and support scientific research transparently. Meta also released LLaMA, a GPT-4 clone with similar aims as OPT.

While OPT and LLaMA are freely available, they cannot be run on ordinary hardware. Training and evaluating such models require dedicated hardware, thus highlighting one of the thesis's significant limitations: inadequate resources and computational power. To train models like RoBERTa-large, an environment like Google Colab[2] with access to a GPU is required. Although GPU access facilitated rapid training of even large models like RoBERTa with 125 million parameters, working with higher-parameter models like OPT with 1.3 billion parameters was nearly impossible. After reducing the batch size to four samples per batch, the available GPUs could barely train the OPT model with 1.3 billion parameters, but training time remained an issue. One training epoch took approximately 30 minutes, contrasting the 3 minutes for RoBERTa-base and 9 minutes for RoBERTa-large. Even after training the OPT-1.3B model for three 30-minute epochs, the results were worse than RoBERTa-large. While model adjustments and Hyperparameter Optimization might have been helpful, the associated costs were prohibitive. Around €300 was spent renting computational power in the cloud environment. While training the model would have been less expensive, developing and testing new hypotheses also demanded resources, leading to considerable costs.

As a result, the decision was made to use smaller yet older models. Fine-tuning large-scale models proved too costly, time-consuming, and outside the thesis's possibilities. Nonetheless, when large models were tested during this thesis, the results did not differ dramatically between RoBERTa-base or -large implementations. In the later process, the OPT model with 1.3 billion parameters was still leveraged during Ensemble Learning and improved the Micro F1-Score from 0.75 to 0.78.

According to Sam Altman, CEO of OpenAI, the era of continuously larger Language Models may be ending [Mil23]. Interestingly, the company's next focus will not be developing larger model architectures with more parameters as previously. This aligns with the thesis that most changes in model architecture during the propaganda detector development were not helpful, and the most remarkable improvement in Propaganda

[1] https://chat.openai.com (last accessed: 29 May 2023)

[2] https://colab.research.google.com (last accessed: 29 May 2023)

Detection would come from creating a better, larger, and more balanced Propaganda Technique Corpus dataset.

6.0.5 Analyzing Russian-language Articles from Russia Today

Analysis of Russian and American news articles revealed a significant difference in length. On average, American articles contain 43.78 sentences, while Russian articles are more than twice as short, with only 19.36 sentences each.

The different news sources made it evident that the articles obtained from the countries varied in their style. For instance, Tass News issues brief press releases focused solely on delivering information. In contrast, Fox News typically publishes longer articles that incorporate entertainment elements and often include the author's perspective.

After searching for another Russian news source, that would show similar characteristics as American news publishers, an interesting observation was made after checking the Russian-language version of Russia Today with its corresponding English-language version. The Russian version would host an extra category named *Opinion*, where Russian authors would write lengthy articles on current topics incorporating their personal opinion.

Since no better alternative was found, those articles were downloaded. In the timespan from January 2022 to April 2023, around 723 articles were obtained this way. Next, the articles were translated from Russian to English with the Marian model [JDGD+18]. The translated articles were handled in the same manner as described in Chapter 4. After running the Propaganda Detection Model on these translated articles, the findings show an average propaganda frequency of 27.50% for those articles. The average sentence count in these translated articles (47.71%) is comparable to that of CNN News (47.71%), ABC News (48.39%), and Politico (50.96%). The actualized overview of Table 5.2 can be found in Table 6.1.

6.0.6 Future Research Directions

Future research could explore alternative methods like Unsupervised Learning to cluster different propaganda techniques rather than relying on Supervised Learning and labeled datasets.

A potential research direction could involve utilizing various propaganda datasets to create more propagandistic spans, labeling the techniques with a weak learner, and reapplying them to train additional models, similar to the method employed by Jurkiewicz et al. [JBKG20].

An untested approach includes incorporating Transfer Learning by utilizing labeled datasets within the disinformation segment or incorporating datasets with sentiment knowledge. Using a Cross-Language Model, the Chinese dataset by Chang et al. [CLCL21] could be utilized. The dataset classifies state-sponsored propagandistic Tweets from China with the techniques proposed by Da San Martino et al. [DSMYBC+19]. Moreover,

Table 6.1: Actualized Overall Analysis of the Articles

Publisher	Scraped Articles	Total Sentences	Detected Propaganda	Detected Percentage	Mean Sentences	Propaganda per Article
ABC News	3998	193472	33362	17.24	48.39	8.34
CBS News	772	26346	5469	20.76	34.13	7.08
CNN News	2986	142043	31041	21.85	47.57	10.4
Fox News	4263	157643	34111	21.64	36.98	8.0
Politico	937	47747	10138	21.23	50.96	10.82
Summary	**12956**	**567251**	**114121**	**20.11%**	**43.78**	**8.81**
News Front	1432	30550	6145	20.11	21.33	4.29
Novye Izvestia	1378	47991	9950	20.73	34.83	7.22
Russia Today RU	20216	395496	91005	23.01	19.56	4.5
Sputnik	7784	158793	37717	23.75	20.4	4.85
Tass News	7647	111819	21244	19.0	14.62	2.78
Summary EN	**38457**	**744649**	**166061**	**22.30%**	**19.36**	**4.31**
Russia Today RU	723	34496	9485	27.5%	47.71	13.12
Summary RU	**723**	**34496**	**9485**	**27.50%**	**47.71**	**13.12**

finally, by utilizing the dataset by Dimitrov et al. [DBAS+21], knowledge derived from propagandistic memes could have been used to enhance the model.

Finally, employing ChatGPT to construct a Propaganda Detector using effective prompts rather than programming represents a promising approach that could make Propaganda Detection accessible to stakeholders with limited technical proficiency.

List of Figures

List of Tables

Acronyms

BERT Bidirectional Encoder Representations from Transformers. 18, 19, 28, 32–36, 55, 90, 92

BIO Begin, Inside, Outside. 44

CRF Conditional Random Field. 25, 44

ELMo Embeddings from Language Models. 21, 34, 35

GPT Generative Pre-trained Transformer. 17

GPT-2 Generative Pre-trained Transformer 2. 17, 28

GPT-3 Generative Pre-trained Transformer 3. 19–21, 92

GPT-4 Generative Pre-trained Transformer 4. 20, 92

GPU Graphics Processing Unit. 21, 92

LIWC Linguistic Inquiry and Word Count. 32, 34

LLaMA Large Language Model Meta AI. 21, 92

LLM Large Language Model. 20, 21

LSTM Long Short-Term Memory Network. 15, 28, 34–36

OPT Open Pre-trained Transformer. 20, 55, 90, 92

RoBERTa Robustly Optimized BERT Pre-training Approach. 18, 19, 28–30, 35, 36, 43–45, 47–49, 55, 90, 92

TF-IDF Term Frequency Inverse Document Frequency. 12, 22, 29, 33, 35

XLM Cross Language Model. 19, 28

XLM-R Cross Language Model RoBERTa. 19

Bibliography

[AAO20] Ola Altiti, Malak Abdullah, and Rasha Obiedat. JUST at SemEval-2020 Task 11: Detecting Propaganda Techniques Using BERT Pretrained Model. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1749–1755, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[AS20] Anastasios Arsenos and Georgios Siolas. NTUAAILS at SemEval-2020 Task 11: Propaganda Detection and Classification with biLSTMs and ELMo. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1495–1501, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[BHL$^+$21] Judit Bayer, Bernd Holznagel, Katarzyna Lubianiec, Adela Pintea, Josephine B. Schmitt, Judit Szakács, and Erik Uszkiewicz. *Disinformation and Propaganda: Impact on the Functioning of the Rule of Law and Democratic Processes in the EU and Its Member States - 2021*. European Parliament, 2021.

[BHR00] Lasse Bergroth, Harri Hakonen, and Timo Raita. A Survey of Longest Common Subsequence Algorithms. In *Proceedings Seventh International Symposium on String Processing and Information Retrieval. SPIRE 2000*, pages 39–48, Los Alamitos, 9 2000. Institute of Electrical and Electronics Engineers.

[BKT20] Verena Blaschke, Maxim Korniyenko, and Sam Tureski. CyberWallE at SemEval-2020 Task 11: An Analysis of Feature Engineering for Ensemble Models for Propaganda Detection. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1469–1480, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[BMR$^+$20] Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language Models Are Few-Shot Learners. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, New York, 2020. Curran Associates Inc.

[Bru01] Ivan Bruha. Pre- and Post-processing in Machine Learning and Data Mining. In Paliouras Georgios, Karkaletsis Vangelis, and Spyropoulos Constantine D, editors, *Lecture Notes in Computer Science*, pages 258–266, Berlin, Heidelberg, 2001. Springer.

[BSA20] Anastasios Bairaktaris, Symeon Symeonidis, and Avi Arampatzis. DUTH at SemEval-2020 Task 11: BERT with Entity Mapping for Propaganda Classification. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1732–1738, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[CCZ06] Olivier Chapelle, Mingmin Chi, and Alexander Zien. A Continuation Method for Semi-Supervised SVMs. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, pages 185–192, New York, 2006. Association for Computing Machinery.

[Cho17] Francois Chollet. *Deep Learning with Python*. Manning Publications, 12 2017.

[CIN20] Anton Chernyavskiy, Dmitry Ilvovsky, and Preslav Nakov. Aschern at SemEval-2020 Task 11: It Takes Three to Tango: RoBERTa, CRF, and Transfer Learning. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1462–1468, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[CKG$^+$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. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online, 7 2020. Association for Computational Linguistics.

[CLCL21] Rong-Ching Chang, Chun-Ming Lai, Kai-Lai Chang, and Chu-Hsing Lin. Dataset of Propaganda Techniques of the State-Sponsored Information Operation of the People's Republic of China, 2021.

[DBAS$^+$21] Dimitar Dimitrov, Bishr Bin Ali, Shaden Shaar, Firoj Alam, Fabrizio Silvestri, Hamed Firooz, Preslav Nakov, and Giovanni Da San Martino. Detecting Propaganda Techniques in Memes. In *Proceedings of the 59th*

Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 6603–6617, Online, 8 2021. Association for Computational Linguistics.

[DCLT19] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, 6 2019. Association for Computational Linguistics.

[DFW20] Guillaume Daval-Frerot and Yannick Weis. WMD at SemEval-2020 Tasks 7 and 11: Assessing Humor and Propaganda Using Unsupervised Data Augmentation. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1865–1874, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[DKS20] Ilya Dimov, Vladislav Korzun, and Ivan Smurov. NoPropaganda at SemEval-2020 Task 11: A Borrowed Approach to Sequence Tagging and Text Classification. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1488–1494, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[DSMBCW$^+$20] Giovanni Da San Martino, Alberto Barrón-Cedeño, Henning Wachsmuth, Rostislav Petrov, and Preslav Nakov. SemEval-2020 task 11: Detection of propaganda techniques in news articles. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1377–1414, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[DSMYBC$^+$19] Giovanni Da San Martino, Seunghak Yu, Alberto Barrón-Cedeño, Rostislav Petrov, and Preslav Nakov. Fine-Grained Analysis of Propaganda in News Articles. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, EMNLP-IJCNLP 2019, pages 5636–5646, Hong Kong, 1 2019. Association for Computational Linguistics.

[DWZ20] Jiaxu Dao, Jin Wang, and Xuejie Zhang. YNU-HPCC at SemEval-2020 Task 11: LSTM Network for Detection of Propaganda Techniques in News Articles. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1509–1515, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[EG20] Vlad Ermurachi and Daniela Gifu. UAIC1860 at SemEval-2020 Task 11: Detection of Propaganda Techniques in News Articles. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1835–1840, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[GBC16] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016.

[GCMK20] Anthony Gillioz, Jacky Casas, Elena Mugellini, and Omar Abou Khaled. Overview of the Transformer-based Models for NLP Tasks. In *15th Conference on Computer Science and Information Systems (FedCSIS)*, pages 179–183, Sofia, 2020.

[GI20] Dmitry Grigorev and Vladimir Ivanov. Inno at SemEval-2020 Task 11: Leveraging Pure Transfomer for Multi-Class Propaganda Detection. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1481–1487, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[HBM$^+$22] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, and others. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022.

[Heb22] Christina Hebel. The Russian Invasion: Putin Settles Accounts with the West. *Der Spiegel*, 2 2022.

[HK00] Arthur E Hoerl and Robert W Kennard. Ridge Regression: Biased Estimation for Nonorthogonal Problems. *Technometrics*, 42(1):80–86, 2000.

[Ho95] Tin Kam Ho. Random decision forests. In *Proceedings of 3rd international conference on document analysis and recognition*, volume 1, pages 278–282, Montreal, 8 1995. Institute of Electrical and Electronics Engineers.

[JBKG20] Dawid Jurkiewicz, Lukasz Borchmann, Izabela Kosmala, and Filip Gralinski. ApplicaAI at SemEval-2020 Task 11: On RoBERTa-CRF, Span CLS and Whether Self-Training Helps Them. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1415–1424, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[JDGD+18] Marcin Junczys-Dowmunt, Roman Grundkiewicz, Tomasz Dwojak, Hieu Hoang, Kenneth Heafield, Tom Neckermann, Frank Seide, Ulrich Germann, Alham Fikri Aji, Nikolay Bogoychev, André F T Martins, and Alexandra Birch. Marian: Fast Neural Machine Translation in C++. In *Proceedings of ACL 2018, System Demonstrations*, pages 116–121, Melbourne, 7 2018. Association for Computational Linguistics.

[JGM20] Yunzhe Jiang, Cristina Garbacea, and Qiaozhu Mei. UMSIForeseer at SemEval-2020 Task 11: Propaganda Detection by Fine-Tuning BERT with Resampling and Ensemble Learning. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1841–1846, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[KB20] Moonsung Kim and Steven Bethard. TTUI at SemEval-2020 Task 11: Propaganda Detection with Transfer Learning and Ensembles. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1829–1834, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[KBG20] Michael Kranzlein, Shabnam Behzad, and Nazli Goharian. Team DoNotDistribute at SemEval-2020 Task 11: Features, Finetuning, and Data Augmentation in Neural Models for Propaganda Detection in News Articles. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1502–1508, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[KGV+14] Subbu Kannan, Vairaprakash Gurusamy, S Vijayarani, J Ilamathi, Ms Nithya, S Kannan, and V Gurusamy. Preprocessing techniques for text mining. In *International Journal of Computer Science and Communication Networks*, volume 5, pages 7–16, 9 2014.

[KGY20] Gangeshwar Krishnamurthy, Raj Kumar Gupta, and Yinping Yang. SocCogCom at SemEval-2020 Task 11: Characterizing and Detecting Propaganda Using Sentence-Level Emotional Salience Features. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1793–1801, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[KMS+19] Johannes Kiesel, Maria Mestre, Rishabh Shukla, Emmanuel Vincent, Payam Adineh, David Corney, Benno Stein, and Martin Potthast. SemEval-2019 Task 4: Hyperpartisan News Detection. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 829–839, Minneapolis, 6 2019. Association for Computational Linguistics.

[KTP20] Anders Kaas, Viktor Torp Thomsen, and Barbara Plank. Team DiSaster at SemEval-2020 Task 11: Combining BERT and Hand-crafted Features for Identifying Propaganda Techniques in News. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1817–1822, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[LBH15] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 5 2015.

[LCG$^+$19] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. *arXiv preprint arXiv:1909.11942*, 2019.

[LMP01] John D Lafferty, Andrew McCallum, and Fernando C N Pereira. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pages 282–289, San Francisco, 2001. Morgan Kaufmann Publishers Inc.

[LWLQ22] Tianyang Lin, Yuxin Wang, Xiangyang Liu, and Xipeng Qiu. A survey of transformers. *AI Open*, 3:111–132, 2022.

[LX20] Jinfen Li and Lu Xiao. syrapropa at SemEval-2020 Task 11: BERT-based Models Design for Propagandistic Technique and Span Detection. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1808–1816, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[MCCD13] Tomas Mikolov, Kai Chen, Gregory S Corrado, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space. In *1st International Conference on Learning Representations*, Scottsdale, 5 2013.

[MCG16] Michael McTear, Zoraida Callejas, and David Griol. Spoken Language Understanding. In *The Conversational Interface: Talking to Smart Devices*, pages 161–185. Springer, Cham, 2016.

[MG13] Neha Mehra and Surendra Gupta. Survey on multiclass classification methods. *International Journal of Computer Science and Information Technologies*, 4(4):572–576, 2013.

[Mil23] Ron Miller. Sam Altman: Size of LLMs won't matter as much moving forward, 4 2023.

[MMOM20] Gaku Morio, Terufumi Morishita, Hiroaki Ozaki, and Toshinori Miyoshi. Hitachi at SemEval-2020 Task 11: An Empirical Study of Pre-Trained Transformer Family for Propaganda Detection. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1739–1748, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[MPS20] Matej Martinkovic, Samuel Pecar, and Marian Simko. NLFIIT at SemEval-2020 Task 11: Neural Network Architectures for Detection of Propaganda Techniques in News Articles. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1771–1778, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[NFR$^+$22] Nic Newman, Richard Fletcher, Craig T Robertson, Kirsten Eddy, and Rasmus Kleis Nielsen. Reuters Institute digital news report 2022. 2022.

[NS16] Krystyna Napierala and Jerzy Stefanowski. Types of minority class examples and their influence on learning classifiers from imbalanced data. *Journal of Intelligent Information Systems*, 46(3):563–597, 2016.

[Ope23] OpenAI. GPT-4 Technical Report. *ArXiv*, abs/2303.08774, 2023.

[PA01] Anthony R Pratkanis and Elliot Aronson. *Age of Propaganda: The Everyday Use and Abuse of Persuasion*. Henry Holt & Co, 2001.

[PBJB15] James W Pennebaker, Ryan L Boyd, Kayla Jordan, and Kate Blackburn. The development and psychometric properties of LIWC2015. Technical report, 2015.

[PCD20] Andrei Paraschiv, Dumitru-Clementin Cercel, and Mihai Dascalu. UPB at SemEval-2020 Task 11: Propaganda Detection with Domain-Specific Trained BERT. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1853–1857, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[PGM$^+$19] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In H Wallach, H Larochelle, A Beygelzimer, F d Alché-Buc, E Fox, and R Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[PNI+18] Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep Contextualized Word Representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, 6 2018. Association for Computational Linguistics.

[Pow11] David M W Powers. Evaluation: From precision, recall and f-measure to roc., informedness, markedness & correlation. *Journal of Machine Learning Technologies*, 2(1):37–63, 2011.

[PP20] Maia Petee and Alexis Palmer. UNTLing at SemEval-2020 Task 11: Detection of Propaganda Techniques in English News Articles. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1847–1852, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[PSA20] Rajaswa Patil, Somesh Singh, and Swati Agarwal. BPGC at SemEval-2020 Task 11: Propaganda Detection in News Articles with Multi-Granularity Knowledge Sharing and Linguistic Features Based Ensemble Learning. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1722–1731, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[PSM14] Jeffrey Pennington, Richard Socher, and Christopher D Manning. GloVe: Global Vectors for Word Representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014.

[RJR+20] Mayank Raj, Ajay Jaiswal, Rohit R.R, Ankita Gupta, Sudeep Kumar Sahoo, Vertika Srivastava, and Yeon Hyang Kim. Solomon at SemEval-2020 Task 11: Ensemble Architecture for Fine-Tuned Propaganda Detection in News Articles. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1802–1807, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[RNSS18] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language understanding by generative pre-training. 2018.

[RS04] Jennifer Rowley and Frances Slack. Conducting a literature review. *Management Research News*, 27(6):31–39, 1 2004.

[RWC+19] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[SDCW19] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *Computing Research Repository*, 2019.

[SF11] Carlos N Silla and Alex A Freitas. A survey of hierarchical classification across different application domains. *Data Mining and Knowledge Discovery*, 22(1):31–72, 2011.

[SP97] Mike Schuster and Kuldip K. Paliwal. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11), 1997.

[SSKM20] Paramansh Singh, Siraj Sandhu, Subham Kumar, and Ashutosh Modi. newsSweeper at SemEval-2020 Task 11: Context-Aware Rich Feature Representations for Propaganda Classification. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1764–1770, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[TLI$^+$23] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. LLaMA: Open and Efficient Foundation Language Models, 2 2023.

[vEH20] Jesper E van Engelen and Holger H Hoos. A survey on semi-supervised learning. *Machine Learning*, 109(2):373–440, 2020.

[VMC20] Ekansh Verma, Vinodh Motupalli, and Souradip Chakraborty. Transformers at SemEval-2020 Task 11: Propaganda Fragment Detection Using Diversified BERT Architectures Based Ensemble Learning. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1823–1828, Barcelona (online), 12 2020. International Committee for Computational Linguistics.

[VSP$^+$17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[WH00] Rüdiger Wirth and Jochen Hipp. CRISP-DM: Towards a Standard Process Model for Data Mining. *Proceedings of the 4th International Conference on the Practical Applications of Knowledge Discovery and Data Mining*, 2000.

[WKW16] Karl Weiss, Taghi M Khoshgoftaar, and DingDing Wang. A survey of transfer learning. *Journal of Big Data*, 3(1):9, 2016.

[Wol92] David H Wolpert. Stacked generalization. *Neural Networks*, 5(2):241–259, 1992.

[XDH+20] Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. Unsupervised data augmentation for consistency training. *Advances in Neural Information Processing Systems*, 33:6256–6268, 2020.

[YDY+19] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V Le. XLNet: Generalized Autoregressive Pretraining for Language Understanding. In *Advances in neural information processing systems*, volume 32, Red Hook, 2019. Curran Associates Inc.

[YtLYc05] Zhang Yun-tao, Gong Ling, and Wang Yong-cheng. An improved TF-IDF approach for text classification. *Journal of Zhejiang University-SCIENCE A*, 6(1):49–55, 2005.

[Zhu08] Xiaojin Zhu. Semi-Supervised Learning Literature Survey. *University of Wisconsin-Madison Department of Computer Sciences*, 12 2008.

[ZRG+22] Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. OPT: Open Pre-trained Transformer Language Models. *arXiv preprint arXiv:2205.01068*, 2022.

[ZWT02] Zhi-Hua Zhou, Jianxin Wu, and Wei Tang. Ensembling neural networks: Many could be better than all. *Artificial Intelligence*, 137(1):239–263, 2002.

[ZWYJ21] Liu Zhuang, Lin Wayne, Shi Ya, and Zhao Jun. A Robustly Optimized BERT Pre-training Approach with Post-training. In *Proceedings of the 20th Chinese National Conference on Computational Linguistics*, pages 1218–1227, Huhhot, 8 2021. Chinese Information Processing Society of China.