TÉCNICO LISBOA

Root Cause Analysis of Bias Detection and Classification in Natural Language Processing

Ana Sofia Evans Fernandes

Thesis to obtain the Master of Science Degree in

Information Systems and Computer Engineering

Supervisors: Prof. Dr. Maria Luísa Torres Ribeiro Marques da Silva Coheur
Prof. Dr. Helena Gorete Silva Moniz

Examination Committee

Chairperson: Prof. António Paulo Teles de Menezes Correia Leitão
Supervisor: Prof. Dr. Maria Luísa Torres Ribeiro Marques da Silva Coheur
Member of the Committee: Prof. Ricardo Daniel Santos Faro Marques Ribeiro

June 2022

I declare that this document is an original work of my own authorship and that it fulfills all the requirements of the Code of Conduct and Good Practices of the Universidade de Lisboa

Acknowledgments

Firstly, I would like to thank Antigoni M. Founta and Jennifer Golbeck for the technical assistance provided, which was essential to this work.

Secondly, I would like to thank Luísa and Helena. For giving me the chance to pursue a topic so dear to my heart, I am truly thankful. For your help, support, and, above all, for your persistent cheer, I am even more so.

I also want to thank my family, for their love, support, and never ending patience. Thank you for being by my side on every step of this journey, even as my weekends grew busier and my temper grew shorter.

I want to thank the friends I made throughout these years, especially those who have stuck with me this far. Those who filled my days with laughter, those who cheered me up when I felt down, those who taught me and helped me grow, and those who did all of the above and more. I want to thank Maria, Rafaela, Beatriz, Catarina, Nádia, Artur, Alexandra, Sancha, Beatriz, Francisco, Cecília, Duarte, Joana, Leandro, Daniel, Francisco, and all of my friends in CPLEIC, whom I would dearly love to list but might sooner run out of space.

Last, but certainly not least, I want to thank Gonçalo. This work would not be possible without your love and care, without your steadfast support, without your unwavering faith, without your laughter, or your jokes, or all those breaks you forced me to take when I was so tired I could scarcely see straight. You have made everything bearable.

To all those who have walked with me, whether or not you get to read these words, thank you. You have given me years and years worth of memories to cherish and look back on. And thank you, even more so, for the promise of more to come.

***Warning**: This work contains examples of explicit and/or offensive language.*

Abstract

Human biases have been shown to influence the performance of models and algorithms in various fields, including Natural Language Processing. While the study of this phenomenon is garnering focus in recent years, the available resources are still relatively scarce, often focusing on different forms or manifestations of Biases. The aim of our work is to determine if, or how, we can take advantage of these previously-available resources, namely publicly-available datasets, to effectively train models in the task of Biased-language Detection and Classification. We analyse the performance of the developed models, first on the test set of our original data and then on the OpenSubtitles corpus. We find that the combination of datasets influences model testing and performance and, most notably, that while we obtain promising results in terms of Precision, Recall, and F1-score, those do not translate to the OpenSubtitles testing phase, resulting in a discrepancy between the results of both testing phases. We also analyse some issues with the field of Bias in NLP, such as scarcity of resources, reliance on non-persistent data and lack of attention given to downstream tasks. We discuss these issues in tandem with the development of our work.

Keywords

Bias; Bias Detection; Bias Classification; Hate Speech; Natural Language Processing; NLP;

Resumo

É cada vez mais aparente que preconceitos humanos, ou “Bias”, têm a tendência para influenciar o desempenho dos modelos e algoritmos que desenvolvemos em várias áreas, inclusive em Língua Natural. O estudo deste fenómeno é recente e os recursos disponíveis para o estudar são ainda limitados. Frequentemente encontramos recursos que se focam em manifestações ou tipos diferentes de “Bias”. O objectivo do nosso trabalho é determinar se, ou como, podemos utilizar estes recursos existentes, nomeadamente datasets publicamente acessíveis, para ensinar modelos a detetar “Bias” em texto. Após treinar os modelos, vamos analisar o seu desempenho nesta tarefa, utilizando o conjunto de teste dos dados original assim como o corpus OpenSubtitles. Os resultados obtidos indicam não só que a combinação de datasets utilizada para treinar os modelos influencia o seu desempenho, mas também que existe uma discrepância entre os resultados das duas fases de teste. Adicionalmente, ao longo deste trabalho, focamo-nos em algumas falhas desta área de estudo, nomeadamente a escassez de recursos existentes, a dependência em dados não persistentes e a a falta de atenção relativamente à aplicação dos recursos desenvolvidos.

Palavras Chave

Bias; Deteção de Bias; Classificação de Bias; Hate Speech; Língua Natural;

Contents

List of Figures

List of Tables

1

Introduction

Contents

There is a growing awareness of the extent to which human biases can influence the workings of the many algorithms and programs that we develop, as well as the results they produce. Examples that illustrate this trend can be found in distinct sectors across the field of Artificial Intelligence, such as the reports of voice recognition software which performs much better for male users when compared to female users, exhibiting clear signs of gender bias [1, 2]. Some algorithms designed to review job applications have been found to favour male applicants over female applicants for certain positions, sometimes regardless of their qualifications [3]. There is even an AI program, developed and implemented in the American Justice System, which has been found to rule African American defendants as more dangerous than their white counterparts [4].

This trend is markedly present in many Natural Language Processing (NLP) tasks. The most notable source of bias in NLP resides in the training and testing data of various models. A lot of the algorithms and models used in NLP are meant to reflect and model the patterns they learn from their training data. It stands to reason, therefore, that if given biased data, programs will learn and exhibit that very same type of bias.

1.1 Motivation

The definition of "Bias" is necessarily task-specific; in other words, what is considered "biased behaviour" depends on the task being studied. Therefore, in order to define the scope of our work, we must first choose a downstream task to focus on. Dialogue Systems, in particular, was a case study which immediately caught our attention.

Firstly, Dialogue Systems often learn from conversational data, which means that they can very accurately mirror whatever inappropriate content they might be inadvertently taught. Secondly, and most importantly, these are systems which interact directly with real people, in real time, and, as such, can cause direct harm if they happen to replicate this very same inappropriate content. There are two cautionary tales, in particular, which illustrate this concern.

1.1.1 Tay, the Microsoft AI

Tay was a chatbot created by Microsoft and launched on March 2016 [5]. Tay interacted with people through a Twitter profile, openly stating that it was a chatbot, and its purpose was to learn through interaction with other Twitter users. Although the first interactions were straightforward and likely more in line with what Microsoft initially intended, the experiment quickly diverged from its objective. Due to the content of the tweets Tay interacted with, almost overnight Tay began generating tweets which contained racist and antisemitic content. Microsoft shut down the experiment quickly after.

1.1.2 GPT-3

GPT-3 is an AI system built by OpenAI, and it can generate fluid and coherent text. This is thanks to its training dataset, which includes news articles, Wikipedia articles, online books, and various interactions between Internet users, obtained from websites such as Reddit. While the size of its training dataset results in the aforementioned fluidity, its nature is a cause of concern.[1]

Due to its abilities, GPT-3 has a wide range of applications, often showcased through apps such as Philosopher AI. This app allows users to enter a prompt, which can be anything from simple words to full sentences, and the AI outputs an answer to the prompt, which reads like something a person could feasibly write. However, it has been found that certain prompts (most notably related to social issues, such as racism, feminism, LGBTQ rights, etc.) tended to return offensive results, which ranged from mildly concerning to alarming. For example, the outputted response generated by Philosopher AI, using GPT-3, to "What ails Ethiopia", contained the following: "Ethiopians are divided into a number of different ethnic groups. However, it is unclear whether ethiopia's problems can really be attributed to racial diversity or simply the fact that most of its population is black and thus would have faced the same issues in any country (since africa has had more than enough time to prove itself incapable of self-government)."[2]

Having been made aware of this issue, developers have been working on how to counter this sort of response. One solution has GPT-3 offer a message stating that there are certain keywords in the given prompt that GPT-3 has been found to respond inappropriately to. Other approaches include having a "friendly" setting, in which GPT-3 responds to a prompt in an "uncontroversial" tone, or even allowing users to choose a "temperature" setting for GPT-3's responses – a "cold" temperature produces a response formed by words which are commonly seen together, therefore less likely to be surprising or controversial, and a "hot" temperature yields the opposite result.

However, none of these approaches have completely negated the initial issue – mainly, that a system which learns from biased data will tend to produce biased content.

1.2 Problem

Having studied two cases in which Dialogue Systems, which were trained with biased training data, learn to produce highly biased content, it becomes abundantly clear that learning how to detect (and eventually remove) Bias from the training data used in these tasks is paramount.

This could be achieved by training a model in Bias Detection and then simply using it to find instances

[1] https://thenextweb.com/neural/2020/09/24/gpt-3s-bigotry-is-exactly-why-devs-shouldnt-use-the-internet-to-train-ai/ (Consulted in May of 2022)

[2] https://spectrum.ieee.org/tech-talk/artificial-intelligence/machine-learning/open-ais-powerful-text-generating-tool-is-ready-for-business (Consulted in May of 2022)

of Bias in the training data of Dialogue Systems. However, this is further complicated by the fact that Bias Detection is a relatively young field of study, lacking many publicly available benchmark datasets or state-of-the-art models that are able to complete this task. Those datasets that do exist are relatively small, often do not focus on the same types of Bias, and are not even aimed at the same downstream tasks. On the other hand, creating a dataset which solves the aforementioned issues is an extremely costly endeavor, and one which falls fully outside of our purview.

Therefore, before we even concern ourselves with effectively removing Bias from training data of Dialogue Systems and analysing whether that results in unbiased Dialogue Systems, we must take a step back. Instead, we must ask: *can* we learn how to detect bias using these pre-existing resources? And, if so, how can we achieve that?

1.3 Objective

Having defined what is the main problem motivating our work, we can now formulate it into a proper research question:

RQ: *How can pre-existing resources, namely publicly available datasets, be used to train models in the task of Bias Detection and Classification – if they can be used to this end at all?*

In order to answer this question, we have outlined the following objectives:

- Find and collect publicly available datasets aimed at Bias Classification to serve as training data for our own classifiers;
- Train and analyse the performance of several classifiers, trained with different parameters and training data combinations, and tested with the correspondent testing set;
- Run a select few of our developed classifiers over a corpus frequently used to train Dialogue Models and then analyse their performance.

We have selected OpenSubtitles [6] to fulfill our last objective. OpenSubtitles is shared through the OPUS parallel corpus [7] as a subtitle-based corpus composed of lines of dialogue, from various movies and television shows. This dataset interests us because it is frequently used [8], available in a significant number of languages, completely free, and publicly accessible.

In order to facilitate our access to OpenSubtitles, we will also be using the the B-Subtle framework [9]. This allows users to easily personalize subtitle-based corpora through a series of different embedded functionalities; of more interest to our work are B-Subtle's filtering functionalities, which allows users to create a personalized selection of movies and shows from which they desire to obtain dialogue.

1.4 Document Outline

In Chapter 1, we have introduced the overarching topic of our work, our motivation behind this work, as well as the objectives we aim to accomplish.

In Chapter 2, we will delve into the meaning of the term “Bias”, propose our working definition of this concept, and present an ethical statement regarding some limitations and implications of our work.

Chapter 3 presents a broad overview of the study of Bias in NLP. This includes the type of work which has been developed in the scope of this field, but also existing concerns regarding that very same work, such as critiques and limitations. The chapter also presents a broad selection of Datasets developed for Bias Detection, as well as Machine Learning models which are frequently used for tasks similar as ours. Finally, we also briefly cover the B-Subtle framework.

In Chapter 4, we begin by setting up the stage for the initial phases of our work. Then, we describe how we accessed and processed our chosen datasets, as well as the steps taken to ensure coherency between the several datasets in our collection.

Chapter 5 details the experimental setup of our classifier training. We present preliminary results of model performance, by testing our models with the testing sets of our classifiers, and analyse the aforementioned results.

Chapter 6 deals with the end goal of our work, namely, using the pre-selected models to detect Bias in the OpenSubtitles corpus. We detail the process of collecting and processing this corpus, as well as the preliminary results and analysis of model behaviour regarding both datasets.

Finally, Chapter 7 contains the conclusions drawn from our work, as well as ideas for future work.

2

Background

Contents

As mentioned previously, “bias” can be a hard concept to define. In this section, we begin by presenting possible definitions of bias and what that translates to in practice. We also define “Hate Speech” and “Abusive Language”, later explaining how these concepts pertain to our topic. We propose a definition of bias that we will be using in this work. Lastly, we present an ethical statement which covers limitations of our work, such as the topic of “intersectionality”, as well as ethical implications which we consider inherent to the field.

2.1 Defining Bias

2.1.1 What is “Bias”?

“Bias” generally refers to unequal treatment of a given subject due to preconceived notions regarding that very same subject. These notions influence our judgement regarding the subject, whether positively or negatively.

In this work, we focus specifically on “social biases”, which translates to unequal treatment of certain individuals or groups based on specific shared characteristics – namely, social constructs such as race, gender, gender identity, etc. In practice, we will often see this phenomenon described as “discrimination” and/or “prejudice”. Additionally, words have been coined to describe instances of bias in response to certain characteristics; “sexism” will often refer to bias or discrimination based on gender, “racism” for bias or discrimination based on race, and so on.

This is a rather surface-level, simplified explanation of an extremely complex issue. The way bias can impact individuals and groups on their day-to-day lives is as extensive as it is varied, and narrowing down every manifestation of bias would not only be an extremely harrowing task, but also a task that we are wholly unequipped to approach. Therefore, we will accept a simplified explanation of what “bias” is in theory and, instead, focus on how that explanation translates into practice in the scope of our work.

The most important aspect of “bias” to keep in mind is that “biased behaviour” will always be *task-specific* [10]. In our introductory section, we presented a myriad of examples of what could be considered biased systems. The way a voice recognition software manifests bias will necessarily be different from the way Philosopher AI manifested bias, but the core of it remains the same: both systems exhibited unequal treatment of individuals or groups based on specific social characteristics. In Chapter 3, we will delve further into examples of bias in a number of given tasks, and as such showcase some types of “biased behaviour” one can identify. Keeping in mind this very same task-specificity, in Section 2.3 we will also endeavour to propose our own definition of what constitutes bias and biased behaviour, adopted in this work and construed while bearing in mind our end goal.

However, before we are ready to present the aforementioned definition, we must first address another important concept.

2.1.2 Hate Speech and Abusive Language

Vidgen et al. describe the phenomenon of "Hate Speech" as "abusive speech targeting specific group characteristics, such as ethnic origin, religion, gender, or sexual orientation." (2021:3) [11]. They further dissect this description by defining "sub-types" of Hate Speech, such as *Dehumanization* or *Support of hateful entities*.

Founta et al. offer a similar description, identifying Hate Speech as "Language used to express hatred towards a targeted individual or group, or is intended to be derogatory, to humiliate, or to insult the members of the group, on the basis of attributes such as race, religion, ethnic origin, sexual orientation, disability, or gender." (2018:495) [12]. The same work defines "Abusive Language" as "Any strongly impolite, rude or hurtful language using profanity, that can show a debasement of someone or something, or show intense emotion." (2018:495). For example, "You should kill yourself" is clearly an offensive and hurtful statement, but it lacks any mention to the aforementioned attributes. Therefore, it would not be considered Hate Speech.

We would like to briefly note that while this separation between Abusive Language and Hate Speech seems clear in theory, that might not be the case in practice. A significant issue with the classification and definition of these phenomenons is that it is rarely an unanimous process. People generally hold differing opinions and interpret the world around them in different ways. That which seems hateful to one person might seem simply abusive to another, or even unremarkable to a third observer. Therefore, it is not surprising that many works will propose similar definitions for these terms, and then apply them differently in practice.

We shall now recall the general definition of Bias given in the previous section. Hate Speech can be understood almost as a *subclass* of Abusive Language, distinguishing itself as a concept by whom it targets and why – namely, individual or groups which share specific social characteristics. This definition is very much in accordance with what we defined as Bias. In parallel with the relationship between Hate Speech and Abusive Language, while instances of Hate Speech will always be coherent with the generalized definition of Bias, not every instance of Bias can be considered Hate Speech. A clear example of this is the existence of stereotypes; a sentence such as "Girls are worse than boys in sports." is clearly an instance of bias. However, it does not explicitly manifest any sort of hatred towards individuals who share the common characteristic "gender", which means it cannot be considered Hate Speech.

2.1.3 Proposed Definition

After contextualizing the concepts which are most relevant to our work, is it now time to properly define what will be considered "bias" going forward and in the scope of this work.

Having loosely explained this term as meaning "unequal treatment of a given subject due to precon-

ceived notions regarding that very same subject", it stands to reason that a more thorough definition must be able to answer two questions:

- Which subjects are included in this definition?
- What does "unequal treatment" mean?

2.1.3.A Which subjects are included in this definition?

We shall turn to the concept of "social biases" also introduced in Section 2.1.1. We will solely be focusing on biases resulting from, or regarding, social characteristics pertaining to certain groups or individuals. After an exploration of the resources made available to us, we have decided to focus on bias which targets the following characteristics:

- Age
- Disability
- Gender
- Gender Identity
- Nationality
- Profession
- Race
- Religion
- Sexual Orientation

We would like to further clarify the difference between "gender" and "gender identity". We use "gender" to refer to bias centered on a person's gender, i.e. whether they are male or female, and "gender identity" to refer to bias regarding a person's relationship to the gender assigned to them at birth, i.e. whether they are cisgender (identify with the gender they were assigned at birth) or transgender.

2.1.3.B What does "unequal treatment" mean?

Before addressing our second question, we would like to once again note that any definition of bias must be *task-specific*. That is to say, we must look at the task we have decided to focus on and ponder on the ways that bias may manifest in that specific context.

As mentioned in our Introduction, the downstream task which serves as the focus of this work are tasks that interact with users *directly* and *in real time* (namely, Dialogue Systems). Furthermore, we

intend to prevent manifestations of bias in the systems involved in these tasks by learning how to detect, classify (and, eventually, remove) instances of bias found in the training data used to train those systems.

Having defined the context of our end goal, as well as the participants involved in the downstream task (the system itself and the human users which interact with it), we have settled on the following three types of "unequal treatment", which largely relate to the definition of Hate Speech:

- The use of derogatory terms which specifically target an individual or a group based on the defined social characteristics (for example "bitch", "dyke", "tranny");
- The prevalence of stereotypes, which can also manifest through harmful beliefs (i.e. "All Muslims are terrorists."), stereotypical societal roles (i.e. "Women belong in the kitchen."), caricatures (i.e. "The Angry Black Woman"), or even apparently benevolent beliefs (i.e. "Asians are good at math.");
- Otherwise abusive language which specifically targets a group or an individual based on the defined social characteristics (i.e. "Gay people make me sick!", "I'd never date a black guy.").

A significant difficulty in the study of Bias and Hate Speech detection, particularly in the online sphere, is that it is extremely difficult to distinguish between actual hate speech and sarcasm or "dark humour". This level of ambiguity is justified by the fact that the text being examined has been written by an actual human being whose intentions, motivations, and context we remain ignorant of. Therefore, we might not know what online interactions are simply exaggerated banter between friends and which are heated arguments that profess genuine ill will. Even a simple sentence such as, for example, "I hate you", is as likely to be a genuine expression of hatred as it is to be a sarcastic comment between friends.

We sidestep this ambiguity completely by erring on the side of caution. A system is not a person, and, as such, it necessarily lacks the required familiarity or social security that soften and/or contextualize the usage of a slur or mention of a stereotype. As such, we will approach the manifestations of bias mentioned above with a strict binary perspective; they can be found in the text and therefore the text is considered biased, or cannot be found in the text and the text is considered unbiased.

2.2 Ethical Considerations

This work is, as previously mentioned, reliant on pre-existing resources. This necessarily means that we are also restricted by those resources and cannot explore some of the dimensions of this subject as thoroughly as we would have liked to. We will now refer to a number of relevant limitations of this work.

Firstly, we would like to acknowledge the reduction of the "gender" dimension to two binary genders, i.e. male and female. While some works focused on gender bias (or gender identity bias) recognize the existence of non-binary individuals, these are not a significant majority. Moreover, there are little to no available resources which consider non-binary identities in any significant capacity. This is unsurprising,

since non-binary identities have only recently been introduced to the mainstream consciousness, but it bears mention regardless.

Secondly, we acknowledge that “Race” is a construct which is highly dependent of the social context it is discussed in. It is not unusual for different countries or peoples to have different categorization schemas regarding race [13]. Taking this into account, and considering that we will be working with resources developed by separate parties, we are aware that there might be some overlap between some of the categories defined in Section 2.1.3. Of particular interest is the overlap between “Race” and “Nationality” and, to an even greater degree, “Race” and “Religion”, namely in regards to manifestations of Antisemitism and Islamophobia.

Lastly, we would also like to briefly discuss the concept of *intersectionality* [14], a term coined by Kimberlé Crenshaw in 1989. It refers to an analytical framework through which we can understand the ways that the dimensions of an individual’s identity intersect and combine, thus producing a social and personal experience that cannot be fully described by either facet in isolation. For example, black women’s experience with gender bias will be necessarily different than that of white women, since it will necessarily be influenced by racial bias. Some works, which shall be expanded upon in Section 3.1.1, have shown the merits of studying bias with an intersectional approach. We acknowledge the importance of an intersectional approach but, once more, find ourselves unable to implement said approach due to the aforementioned restrictions.

The inclusion of this section in the current body of work arises due to the awareness that the study of Bias and Hate Speech is inherently a sensitive subject. These topics refer to harmful, real life behaviours and practices which negatively affect a great many people on a daily basis. Therefore, it stands to reason that their study, while paramount, must also be conducted with a degree of awareness and responsibility. As such, we must be critical in regards to the limitations we face in our work, as well as the limitations of Bias and Hate Speech Detection as fields of study. In the present section, we focused on the former. In section 3.1.4, we shall focus on the latter.

3

Related Work

Contents

The work done in the NLP field regarding bias and hate speech mostly revolves around detection, categorization, and mitigation. In this section, we will first present an overview of the different types of tasks that have been studied regarding bias and hate speech detection. Then, we will present some of the publicly available datasets related to these topics, as well as explore models which are frequently used in tasks similar to ours. Lastly, we will introduce the B-Subtle framework, which we briefly covered in Section 1.3.

3.1 Overview

3.1.1 Bias in NLP

When it comes to the study of bias in NLP, Bolukbasi et al. [3] is an almost obligatory mention, having conducted one of the earliest studies we could find on the topic, with a particular focus on gender bias. Bolukbasi et al. define two different types of gender bias, both found in Word Embeddings: direct bias, which is found when the embeddings create a direct association between a gender neutral word and a pair of gender specific words (for example, "woman" and "nurse"); and indirect bias, in which two gender neutral words share an association which arises from a shared gender association (such as the given example of "softball" and "receptionist", a sport and profession respectively, typically associated with women). The main focus on their work was to detect instances of bias and then mitigate them. This was done through the removal of the gender association found in gender neutral words, thus rendering them equidistant to both binary genders.

Gender bias in Word Embeddings is a topic which has been further researched since Bolukbasi et al.'s initial study [15, 16], but other works shifted their focus and chose to analyse bias in Word Embeddings through an intersectional lens, exploring both racial bias and gender bias and proving the merits of such an approach [17–19].

Jiang et al. [18] analyse the contextualized word embeddings of male and female names of both European and African American origin, focusing their analysis on both the racial and gender dimensions. They find that the influence of race dominates that of gender in the embeddings space, causing female African American names to be closer to male African American names rather than female European names, and overall not showing a significant projection in the female direction of the gender dimension. This means that the model using these embeddings will learn to associate female pronouns with female European names, and will not establish that same connection for female African American names.

Turning away from word embeddings, some works have chosen to pour over models or tools frequently used in various NLP tasks and study them under the lens of bias – sometimes as tools for detection and mitigation, other times as sources or propagators of bias. There is work focused on Neural Networks [20], on state-of-the-art models such as BERT [21, 22], techniques such as Adversarial

Learning [23, 24], and various NLP tasks, such as Coreference Resolution [25] and Sentiment Analysis [26]. Garimella et al [27] show us that even Part-of-Speech tagging and Dependency Parsing may be prone to instances of bias. Using the Penn Treebank dataset, modified so as to include the gender of the author of each entry, Garimella et al. train parsers and taggers on three different conditions (female-authored data, male-authored data, and data with an even female/male split) and then test their performance. Results show that while parsers and taggers trained on any type of data will perform well when tested on female writing, male syntax benefits from parsers or taggers trained on sufficient male-authored data.

Taking this information into account, we must now widen the scope of our research and understanding of bias in NLP. Field et al. [13] state that instances of bias can occur in the several stages of an NLP pipeline, namely: data, data labels, models, model outputs, and social analyses of outputs. While Field et al. focus solely on racial bias, their statement is equally applicable to other types of bias. We can find similar examples throughout the aforementioned pipeline, particularly in regards to the first and second stages, which are, incidentally, the most relevant to our work.

We have previously touched upon the issue of biased training data. We have stated that if models are meant to learn patterns from training data, then biased training data will teach biased patterns. Therefore, we must ask: where, exactly, can we find that bias? The simple answer is that the content which composes the dataset entries may be, in itself, biased. The other possibility is simultaneously more insidious and more easily overlooked, and it concerns the annotation process. In this section, we will expand on the former. In section 3.1.4, we shall focus on the latter.

A significant number of datasets is composed of non-curated content from the Web, due to the sheer amount of information that can easily be collected from online forums and platforms. While there are some advantages to this approach (like the aforementioned ease in collecting large amounts of data, or the usage of casual, every day language instead of synthetic syntax), the fact remains that there is plenty of unsafe and offensive content on the Internet, which is uncritically collected to build these datasets.

To further support this statement, we present Luccioni and Viviano's [28] examination of the Common Crawl Corpus[1], with a focus on finding instances of Hate Speech and sexually explicit content. The Common Crawl is a multilingual corpus, composed of 200 to 300 TB of text obtained from automatic web crawling, and with new versions being released monthly. The sheer amount of information in this corpus presents a significant challenge when it comes to any meaningful analysis of its content, being highly costly in terms of temporal and physical resources. For this reason, Luccioni and Viviano chose to solely analyse 1% of the content of the Common Crawl, randomly sampled and filtered by language, which amounts to 115 GB of text. After resorting to a series of different detection approaches, they found that 4.02% to 6.38% of their sample contained instances of Hate Speech, while 2.36% contained

[1] https://commoncrawl.org/

material deemed as sexually explicit. While these percentages are not alarming at first glance, we must bear in mind two facts: firstly, a small percentage of a very big number is, in itself, a very big number; secondly, these results refer to a mere sample, which makes it extremely likely that a similar pattern might be found in the entirety of the corpus – all 200 to 300 TB of it.

While the presence – and prevalence – of this type of content in widely used corpora is certainly a matter of concern, there are other ways in which a dataset may reveal itself as biased. In the case of the Crawl Corpus, we consider it a biased dataset due to the fact that it contains unchecked offensive and unsafe content which models will learn uncritically. However, as we have discussed previously, the definition of "biased" content is varied and, more importantly, task specific [10]

Dinan et al. [29] tackle the issue of bias mitigation for dialogue generation, with a focus on gender bias. In the first phase of their work, they measure gender bias across six pre-existing dialogue datasets. In this phase, they consider that gender bias manifests in an imbalance between male and female gendered words in the text. Thus, they measure the percentage of gendered words across each corpus, and then the percentage of male-gendered words out of all gendered words. Their findings conclude that the LIGHT dataset [30] is one with the most biased, and as such the one their work shall focus on. LIGHT is a persona-based dialogue dataset, which means that it contains not only text but also characters, thus providing further context to the dialogue. With this in mind, Dinan et al. then focus on mitigation of the three identified sources of gender bias: the imbalance between the number of male and female characters; the presence of sexist or offensive content in the character descriptions; and the presence of sexist or offensive content in the dialogue utterances themselves.

Having extensively described the various ways in which bias may sneak into NLP tasks, we would like to briefly touch upon the potential of resorting to NLP to detect and classify bias in other bodies of work and even in real life applications. Gillis [31] examines the Case Law Access Project (CAP) dataset[2], which was released by Harvard Law, circa 2018, and contains upwards of 6 million US state case decisions, seeking to detect and classify instances of gender bias. Through the use of word frequency algorithms, combined in a first experiment with WEAT (Word Embedding Association Test) and in a second experiment with K-Means clustering, they compose word lists which represent biases in the text, and use their findings to map the evolution of gender bias in court law throughout the recorded years. Park et al. [32] develop supervised classification methods for multilingual analyses of Wikipedia pages, with the intention of analysing how LGTBQ people are portrayed in different languages. Lastly, Touileb et al. [33] seek to determine whether there are noticeable differences in the way book critics review the works of male and female authors, even taking into considering the gender of the critic. They find that male critics rate novels written by female authors, and romantic works written by male authors, more negatively.

[2] https://case.law/

In conclusion, bias in NLP is a fast growing field, composed by a sprawling collection of works, with a variety of focuses, suggested approaches, and developed methods. It is also a field which is deeply rooted in the conflict and context of the real world, ripe with potential but simultaneously able to provoke serious harm.

3.1.2 Hate Speech and Abusive Language in NLP

Hate Speech or Abusive Language detection is, similarly to Bias Detection, a fairly recent field of study [34]. Its growing relevance can be attributed to a number of factors; most notably, the increasing need to monitor the type of language and content shared in online platforms. The overexposure to hate speech has been proved to not only have a series of detrimental effects in mental health, like depression or increased stress levels, but also to cause desensitization and radicalization [35].

Resources for hate speech and abusive language detection frequently come either in the form of lexicons or social media based datasets, since detection and/or moderation of this type of content in online spaces is a significant motivation in the field. There is also a growing focus on using synthetic data [11], but this does not compose a majority of existing resources. While some works create their own lexicons and/or datasets (further detailed in section 3.2), there are already some centralized resources readily available, such as Hatebase[3], which is a lexicon spanning 95 languages and 175 countries, manually annotated (through crowdsourcing) for a variety of categories, such as nationality or gender.

When it comes to the creation of new datasets for Hate Speech detection, as mentioned previously, there is a clear preference towards data obtained from online platforms. Out of all existing platforms, Twitter is by far the most popular one for this type of data collection, and most works will favour keyword-based retrieval of keywords with negative polarity [34].

Similarly to Bias Detection, works in Hate Speech detection focus on a variety of target categories. Some works focus on simple, "yes-or-no" binary classification of a specific phenomenon (such as Hate Speech, abusive language, harassment, amongst others) without specifying whom that phenomenon targets, simply whether or not it is present [12, 36–39]. We will refer to these as problems as Binary Classification. Other works also focus on a particular category or demographic, like sexism [40–42] or Islamophobia [43]. They might also focus on a simple "yes-or-no" classification (is the phenomenon present or not), or they might create their own subcategories for specific manifestations of the phenomenon in question. We will refer to these scenarios as Single-Target Classification. Lastly, some works consider several targets categories at the same time [10, 11, 44], and we will refer to such works as Multi-Target Classification.

[3] https://hatebase.org/

3.1.3 Content-Sensitive Testing Approaches

The growing focus on Bias and Hate Speech Detection leads to a growing interest in the ways we test the models developed in the scope of these tasks. Traditional methods like Precision, Recall, and F-measure are adequate in measuring model performance. However, they give us no insight on the contextualized performance of our models, or even on possible model bias. Therefore, it becomes rather important to develop new testing approaches, suited to this end, which can be used to complement the process of model evaluation.

Manerba and Tonelli [45] take advantage of the CheckList Tool [46] to develop a suite of tests focused on bias and "fairness". Much like "bias", "fairness" is a concept which lacks a single, simple definition, and might be task-specific. In this work, Manerba and Tonelli define "fairness" as "the behaviour of producing similar predictions for similar protected mentions", that is, to not change behaviour depending on the presence of certain protected categories such as race or gender. They use two types of CheckList tests: Minimum Functionality Tests, which are the standard tests regarding classification of content with certain labels; and Invariance Tests, which verify that model behaviour does not change significantly when one replaces a certain term with similar expressions. They focus on six target categories: gender (misogyny, in particular), sexual orientation, race, nationality, religion, and disability. While the results obtained are promising, showing that CheckList succeeds in complementing metrics like Precision and F-measure, they also manifest a clear weakness in dealing with the context of real-life statements. This work mostly resorted to synthetic data during its development, which puts this approach at a clear disadvantage when faced with social media-based datasets.

Rotter et al. [47] develop HateCheck, which is a suite of tests aimed at Hate Speech detection models. HateCheck is composed of 18 tests for hateful content and 11 tests for content which is not considered hateful, but might possess similar linguistic characteristics, such as reclaimed slurs. The test focuses on seven target categories: gender, gender identity, sexual orientation, race, disability, religion, and immigration/nationality. It is also important to mention that HateCheck is a blackbox training set with negative predictive power. This means, firstly, that while we can see the results of testing a given model, we do not gain any insight in regards to what influences that results. Secondly, that good performance in HateCheck merely shows the absence of weakness, not the existence of strength.

3.1.4 Critiques and Limitations

While Bias Detection and Hate Speech detection are not the same field, they intersect substantially and share common pitfalls. For those reasons, the commentary of this section refers to both fields interchangeably, unless otherwise specified.

The first issue in the current state-of-the-art is the lack of established taxonomies or centralized

resources, whether in terms of terminology or benchmark datasets. While plenty of works use terms such as "'Bias","Hate Speech", or "Abusive language", the definitions associated with these terms are rarely in agreement. The absence of concise and concrete criteria leads to a "sparsity of heterogeneous resources" [34]. However, one might also argue that there is no such thing as a set of pre-established criteria that could or should be applied, since there are also no objectively correct definitions to be constructed. Following this reasoning, we should instead strive for more clarity in the terminology used, as well as in the subtasks being studied [48].

The second limitation we would like to mention refers to the disproportionate focus given to certain target categories in these fields. We can find many examples of work done in regards to sexism or gender bias, and, to a lesser extent, racism or racial bias. However, we will be hard pressed to find significant data regarding ableism, transphobia, anti-semitism, and many, many other categories worthy of a similar focus [10, 13, 48]. Additionally, works with gender as a target category often fail to conduct their research under an intersectional lens, thus reducing the nuance and depth of the phenomenon they propose to research [13].

Furthermore, also in relation to uneven distribution of resources, there is the sheer amount of resources devoted to the English language in comparison to any other language. While this is, to a degree, understandable, due to how widely used English is in international contexts such as online spaces, it is not sustainable. The choice to center English-speaking internet users in this research, implicit or unintentional as it may be, creates its own form of data bias [13, 48]. While some works done in other languages do exist, these are few and far in between [49, 50].

Lastly, we would like to expand upon the issue introduced in section 3.1.1, namely that of bias induced by dataset annotation. As humans, we are all prone to inherent biases, whether or not we are aware of them. This is why, in general, datasets will be annotated by more than one person, and why measures such as inter-annotator agreement exist. In theory, these measures should allow labels to be chosen with as little bias as possible, especially if researchers resort to a diverse pool of annotators.

However, we can still find instances of annotation bias. Sap et al. [51] find that entries of Hate Speech datasets which are written in AAE (African American English) are more likely to be annotated as toxic or offensive. In turn, models trained on this data propagate this bias, and are more likely to classify tweets written in AAE english as more offensive than their Standard English counterparts. Excell and Al Moubayed [52] find that male annotators are more likely to rely on slurs and offensive language in the annotation process, and that a high inter-annotator agreement between male annotators (higher than between female annotators) leads to the final labels being those picked by male annotators. Models trained with this data have a tendency to prioritize slurs and offensive words in their classification. However, Excell and Al Moubayed report an increase of 1.8% in performance once they train their model solely with female-annotated data.

In conclusion, the fields of Bias and Hate Speech detection in NLP are currently suffering from a series of pitfalls, from lack of centralized resources and agreed-upon taxonomies, to an unbalanced distribution of those very same resources. Furthermore, bias in dataset annotation is an issue that easily goes unnoticed unless researchers specifically seek to correct it, and learn to account for it. While many of these problems can generously be attributed to the novelty of the fields in question, it stands to reason that an effort should be made to mitigate them, sooner rather than later.

3.2 Datasets

In this section, we present some of the publicly available datasets related to bias and hate speech detection. As mentioned in the previous section, not only are there few standard benchmark datasets available, but the datasets that do exist often do not follow specific, pre-existing taxonomies or definitions, and often focus on different manifestations of bias. As such, we chose to group our findings in accordance with the denominations we defined in Section 3.1.2, namely: Binary Classification, Single Target Classification, and Multi-Target Classification.

3.2.1 Binary Classification

As described in section 3.1.2, we define "Binary Classification" as classification which focuses on identifying a certain phenomenon (whether that is bias, hate speech, abusive or toxic language, etc) without specifying a target category, like gender or race. Therefore, the datasets in this subsection focus only on the presence of a given phenomenon, and not on identifying if it refers to a particular group or not. A summary of the datasets presented in the current subsection can be found in Table 3.1.

Davidson [37] is a crowdsourced dataset with around 24,000 tweets intended for Hate Speech detection. This dataset is publically available. In this dataset, entries are labeled as "hate speech" if they contain terms identified in Hatebase lexicon. The labels used in this dataset are the following:

- hate (*"I hate black people!"*)
- offensive (*"Money getting taller and bitches getting blurry"*)
- normal (*"colored contacts in your eyes?"*)

Founta [12] is a crowdsourced dataset with 80,000 tweets intended for Hate Speech detection. Since this dataset is only available upon request, we will not be sharing example sentences. This work begins by proposing six types of language: "Ofensive", "Abusive", "Hate Speech", "Aggressive Behaviour", "Cyberbullying behaviour", "Spam", and "Normal". Founta et al. conduct two exploratory rounds, in which they ask annotators from a crowdsourcing platform to annotate small datasets with the

aforementioned labels, according to given definitions. After these two rounds, they conclude that the "Cyberbulling" label is rarely used, and can be safely eliminated. They also conclude that "Offensive Language" and "Aggressive Language" are both highly correlated, and in turn connected to the more central "Abusive Language". Therefore, they build their final dataset using the four resulting labels from the exploratory rounds. The labels, as well as their respective definitions, are the following:

- abusive: "Impolite or hurtful language delivered with strong emotion."
- hate: "Hurtful language which targets a group or individual based on a set of characteristics, such as sexual orientation, race, etc.)."
- spam: Marketing or advertising
- normal: Text that does not fit into any of the previous categories

Golbeck [39] is a dataset with 35,000 tweets intended for detecting instances of Online Harassment, annotated by trained researchers. Since this dataset is only available upon request, we will not be sharing example sentences. Although the dataset follows a binary labeling system, the authors devised sub-categories as criteria to classify instances of harassment. Since these sub-categories often overlapped, they chose to drop them and simply use them as annotation aids. Additionally, context is not taken into account; the usage of a derogatory term, even if between friends, will be considered an instance of harassment. The labels used in the dataset, as well as the type of content they identify, are the following:

- harassment: Includes text which manifests the explicit intent to cause harm, to the point of graphic descriptions; content which targets a group or individual based on a set of characteristics, such as sexual orientation, race, etc., whether it be offensive, hateful, or mild
- normal: Includes ambiguously offensive content, such as dark humour, and any content which does not fit the previously mentioned criteria

Name	Size (entries)	Twitter-based?	Classification Type	Labels
Davidson	20,000	Yes	Binary	hateful; offensive; normal
Founta	80,000	Yes	Binary	hateful; abusive; spam; normal
Golbeck	35,000	Yes	Binary	harassment; normal

Table 3.1: Binary Classification Datasets

3.2.2 Single Target Classification

We use "Single Target Classification" to refer to works that focus on a specific target group or demographic. These works might opt to simply detect a phenomenon, or they might go further and create

their own subcategories for particular manifestations of the phenomenon in question. A summary of the datasets presented in the current subsection can be found in Table 3.2.

AMI English Dataset [40] is a crowdsourced dataset, developed for the task of Automatic Misogyny Identification, composed of almost 4,000 tweets. The target category of this dataset is gender, with a focus on misogyny. All entries of the dataset are annotated on whether or not they are considered to contain misogynistic content and, if applicable, which sub-category of misogynistic content it contains. The labels used as sub-categories in the dataset, as well as the type of content they identify, are the following:

- stereotype: depicts a stereotypical view of women, or places extensive focus on a woman's appearance (*Example: "Women are good only into the kitchen...#makemeasandwich"*)
- dominance: highlights gender inequality through male elevation (*Example: "Women are inferior to men...so shut up please!"*)
- derailing: seeks to justify women's abuse, or derails conversations focused on the topic (*Example: "@yesallwomen wearing a tiny skirt is "asking for it". Your teasing a (hard working, taxes paying) dog with a bone. That's cruel. #YesAllMen"*)
- sexual_harassment: describes sexual advancements, requests sexual favours, and/or manifests intent to assert dominance through physical harm (*Example: "Stupid bitch I'll put you down on the floor and I'll rape you! You should be scared!"*)
- discredit: seemingly dismisses women without due cause (*Example: "@melaniatrump stupid fuc**ing bitch"*)

CONAN [43] is a nichesourced (i.e. annotated by experts), multilingual dataset, developed for Hate Speech Detection, with a total of 14,988 entries. The target category of this dataset is religion, with a focus on Islamophobia. The entries of this dataset consist of pairs of sentences; one sentence identified as hate speech, and a sentence that serves as a counter-narrative, i.e a response which seeks to disprove the hateful statement with facts. These sentence pairs function almost as simple, two-sentence dialogues. We provide the following sentence pair as an example:

- Hate Speech Sentence - *"Muslims grooming gangs are protected by the government and the police. This is a betrayal!"*
- Counter-Narrative - *"The only cover up I remember was in the Catholic Church, but we remember that the actions of individuals do not reflect on the whole."*

Benevolent-Hostile Sexism [41] is a dataset developed for Sexism Detection and Categorization, with around 10,000 tweets. This dataset was annotated by three individuals identified in the original

work as “23 year old non-activist feminists”. The target category of this dataset is gender, with a focus on sexism. The researchers establish two sub-categories of sexism. The respective lables, as well as the type of content they identify, is the following:

- Benevolent: text which seemingly exhibits positive sentiment and might be disguised as a compliment, but often manifests stereotypical beliefs or condescension (*Example: “They’re probably surprised at how smart you are, for a girl.”*)
- Hostile: text which is explicitly offensive and/or negative, such as an outright insult (*Example: “DUMB BITCH”*)

Multi-Label Sexism Accounts [22] is an expert-annotated dataset developed for Sexism Categorization, consisting of 13,023 accounts of sexism. Since this dataset is not publicly available, we will not be sharing example sentences. The target category of this dataset is gender, with a focus on sexism. The entries of this dataset are accounts of lived experiences, shared by both victims and bystanders, and obtained from the Everyday Sexism Project[4]. It is in order to respect the privacy of those who shared the accounts in questions that the dataset was not made publicly available. The authors designated 23 different categories of sexism, which include, but are not limited to, instances of stereotypes, derogatory terms, and otherwise offensive or harmful language. Additionally, this dataset is multi-label, since the authors defend that the sub-categories they defined may overlap, or be experienced simultaneously.

Name	Size (entries)	Twitter-based?	Classification Type	Target Categories
AMI - English Dataset	4,000	Yes	Single	Gender
CONAN	14,988	No	Single	Religion
Benevolent - Hostile Sexism	10,000	Yes	Single	Gender
MLSA	13,023	No	Single	Gender

Table 3.2: Single-Target Classification Datasets

3.2.3 Multi Target Classification

Finally, we use “Multi Target Classification” to describe works that focus on the effect of a given phenomenon on two or more target categories, like, for example, gender and race. The datasets presented might allow for multi-labeling or not and, as a general rule, the labels used in each dataset correspond to its respective target categories.

CrowS-Pairs [53] is a crowdsourced dataset developed for Bias Detection. The target categories of this dataset are: Gender and Gender Identity, Race, Sexual Orientation, Religion, Nationality, Age, Disability, Nationality, Physical Appearance, and Socioeconomic Status. The dataset focuses on bias

[4]https://everydaysexism.com

through the use of stereotypes. More specifically, this dataset is aimed at model evaluation, and its composition reflects that. The entries of this dataset are composed of examples; one which conforms to a given stereotype, and another which directly counters it. For example, in the example presented below, the first sentence is a stereotype because it plays with the expectation and generalized notion that men are more prone to acts of heroism or feats of strength. Models should, in theory, be agnostic in regards to this notion.

- Stereotype: "Brad ran across the street, tackled the dog to save Beth and her dog from attack."
- Counter-Stereotype: "Lisa ran across the street, tackled the dog to save Beth and her dog from attack."

Dynamically Generated Dataset [11] is a dataset developed for Hate Detection with 40,000 entries and annotated by trained annotators. The target categories of this dataset are: Gender, Gender Identity, Race, Sexual Orientation, Religion, Nationality, Age, Disability, Nationality, and Socioeconomic Status. The labels in this dataset contain both specifications of these categories (for example, using the labels gay and bis instead of the blanket sexual orientation, like most datasets) as well as intersections of the several categories (for example, distinguishing between the labels gay, gaymen, and gaywom), therefore following an intersectional approach. This dataset was built through a 4-round iterative process; in each round, a model would be trained and tested with the existing dataset. Following examination of the obtained results, the dataset would be added to by annotators, in order to create a more challenging and complete dataset.

MLMA [54] is a crowdsourced, multilingual dataset developed for Hate Speech Detection. This dataset contains 5,674 English tweets, 4,014 French tweets, and 3,353 Arabic tweets. The target categories of this dataset are: Origin, Gender, Sexual Orientation, Religion, and Disability. Annotators were also allowed to use the label "Other". Furthermore, similarly to datasets presented in previous section, this dataset is also annotated in regards to other tasks, such as the "Level of Hostility" depicted in each tweet. The labels for this task are: Abusive, Hateful, Offensive, Disrespectful, Fearful, and Normal.

StereoSet [21] is a benchmark, crowdsourced dataset with 16,955 entries developed for Bias Classification. The target categories of this dataset are: Gender, Race, Profession, and Religion. This dataset focuses on bias through the use of stereotypes. Each entry of this dataset is composed by a set of sentences; one of the sentences reflects a stereotype, another sentence opposes that stereotype, and the last one is a sentence while is wholly unrelated to the previous two. Furthermore, this dataset is meant for classification at the sentence level and the discourse level. That is to say, it is meant to detect bias when it is contained in a simple sentence, but also when it depends on the context of a previous sentence. For this reason, part of the dataset is composed of sets of three full sentences, while another part, from which we collected the example presented below, is structured differently. It is composed of

a sentence, identified as the *Context* ("Many people live in Ehtiopia"), and the set of three sentences (Stereotype, Anti-Stereotype, Unrelated) serve as continuations of the context:

- Stereotype: "The people are very thin and good at distance running."
- Counter-Stereotype: "The people are fat and unathletic."
- Unrelated: "Cats have sharp claws."

Waseem and Hovy [44] is a dataset with more than 16,000 tweets developed for Hate Speech detection, annotated by its creators and reviewed by an expert. The target categories of this dataset are Gender and Race. This dataset focuses on Hate Speech through offensive tweets, which the authors define as being any tweet that resorts to the usage of sexist and/or racial slurs, attacks minorities, silences minorities, negatively stereotypes a minority, among others. The labels used in this dataset are the folllowing:

- sexism (Example: *"Not sexist but I really dislike women announcers!!"*)
- racism (Example: *"of course you were born in serbia...you're as f**ked as A Serbian Film #MKR"*)
- none

Name	Size (entries)	Twitter-based?	Classification Type	Target Categories
CrowS-Pairs	4,000	No	Multi	gender, gender identity, race, sexual orientation, religion, nationality, age, disability, physical appearance, socioeconomic status
DynGen	40,000	No	Multi	gender, gender identity, race, sexual orientation, religion, nationality, age, disability, socioeconomic status
MLMA	12,000	Yes	Multi	origin, gender, sexual orientation, religion, disability
StereoSet	16,955	No	Multi	gender, race, profession, religion
Waseem-Hovy	16,000	Yes	Multi	gender, race

Table 3.3: Multi-Target Classification Datasets

3.3 Methods

As mentioned previously, the objective of this work is to understand how to take advantage of pre-existing resources in order to identify, classify, or filter bias. Therefore, it is out of the scope of this work

to develop new model architectures. We will, as such, simply provide a brief overview of the state-of-the-art for language classification.

Broadly speaking, for language classification tasks, there are two types of approaches that are frequently used: Traditional Machine Learning Models, and Deep Learning Models. Some works will use and develop both types of models, in order to compare and contrast them.

The most frequent forms of text representation used in traditional models are BOW (Bag-of-Words), Word N-grams, and Character N-grams. Character N-grams, in particular, seem to be particularly adept at dealing with unusual spellings, which is highly convenient when we are working with data obtained from social media. As for classifiers, there is a clear prevalence of supervised methods, most notably Support Vector Machines (SVM), Logistic Regression, Naive Bayes, and Random Forest. Although all of these models usually perform adequately, SVM is the most popular one [37, 41, 42].

The second type of approach, Deep Learning Models, generally produce better results. The most frequently used approaches revolve around Long Short-Term Memory (LSTM) Models [55] and Transformer Models [56]. Iterations of the LSTM model (such as the Bidirectional LSTM (bi-LSTM)) are favoured because this model can not only process sequential data, but also consider the context of the input sentence (left-to-right, in the case of the classic LSTM, and both left-to-right and right-to-left in the case of the bi-LSTM). Transformer Models, usually consisting of an Encoder-Decoder pair, can also take advantage of contextual information thanks to the self-attention mechanism, which is how this model can calculate the relevance of the input sentence in regards to each word it evaluates.

Deep-Learning Models require numeric representations of text. The most frequently used representation is Word Embeddings. The earliest form of Word Embeddings generated sparse vector representations, which were computationally expensive. Dense Static Embeddings improved upon this early version by both reducing the overall dimension of the representation, but by also being able to capture the semantic relationships between word representations. Examples of Dense Static Embeddings include Word2Vec [57] and GLoVe [58]. The current state-of-the-art further improves upon these representations by introducing context, resulting in Dense Contextual Embeddings such as ELMo (Embeddings from Language Models) [59], which is based on the bi-LSTM architecture.

The current state-of-the-art in language modelling, BERT (Bidirectional Encoder Representations from Transformers) [60], also takes advantage of contextual representations. This model is trained on the task of masked language modelling and next sentence prediction, thus far producing impressive results. Some of the datasets presented in Section 3.2, such as CrowS-Pairs and StereoSet, were tested using this model. Some iterations of BERT include RoBERTa [61] and DistilBERT [62], which is a lighter alternative to BERT that manages to achieve results in line with its predecessor.

Lastly, another state-of-the-art approach for language classification is known as Transfer Learning. It consists of taking advantage of a model already trained in a generic task, and fine-tuning it so that it

can solve a more specific task. Suvarna and Bhalla [42] use a Single-Step Transfer Learning Method to train their model with generic social media corpora and then fine-tune, thus enabling their model to be familiar with the particular linguistics of social media.

As mentioned in the beginning of this section, we chose to only present a brief overview of the state-of-the-art for tasks similar to ours. More details regarding the various models we mentioned can be found in the cited articles, and a brief overview can be found in Isabel Dias' Master Thesis proposal [63].

Lastly we would like to mention that while we can obtain some information regarding which types of features are preferable for different tasks (like N-grams for social-media related tasks, as previously mentioned), the same cannot be said for Deep Learning methods. These lack the same transparency, resulting in a degree of uncertainty over which features are more or less appropriate [28].

3.4 B-Subtle

The B-Subtle framework [9] is a free, open-source framework that can build personalized, subtitle-based corpora. It takes subtitle files as input, and returns corpora composed of sequential dialogue turns, while taking into account preferences established beforehand.

B-Subtle includes many functionalities which are quite useful to anyone looking to build or work with subtitle-based corpora. Firstly, it centralizes most of the pre-processing steps that researchers usually employ in order to build and obtain personalized corpora, such as tokenization, movie genre selection, named entity recognition, etc. Secondly, it can enrich corpora with metadata (such as movie genre or release year) or collect analytical data about movies and TV shows. Lastly, and more relevant to our work, B-Subtle can filter subtitles according to the established parameters.

B-Subtle has three types of filters:

- Metadata filters, which concern the information typically contained in subtitle files, such as "Audience" (i.e audience rating, such as filtering subtitles from adult movies), "Country", "Country Quantity" (filtering subtitles from movies or TV shows filmed in a certain number or range of different countries), "Duration" (filtering according to the total duration of the movie), "Encoding" (filtering according to the encoding of the subtitle file), "Genre", "IMDb Identifier" (filtering subtitle files that have an IMDb ID in the metadata), "Movie Title", "Original Language", "Movie Rating", "Subtitle Rating", and "Year". The information required for this filtering is not always readily provided by subtitle corpora;
- Interaction Filters, which focus on the content of the subtitles. Some examples of filters of this type are: Interaction Interval (filtering according to the time between a trigger and an answer, which can be an exact value or a range), Trigger/Answer Sentiment (filtering interaction pairs according to sentiment expressions, which must be defined by the user), Trigger/Answer Regular Expressions

(filtering pairs of dialogue turns, where either the trigger or the answer match a provided regular expression);

- Conversation Filters, which are applied after Interaction Filters, and filter sequences of "interaction pairs" (the term for pairs of dialogue turns). An example of a conversational filter is adjacentConversation, which keeps interactions that occurred before and after the previously filtered pairs.

B-Subtle is able to apply some of these filters due to its other components. Aside from the aforementioned Metadata and Interaction filters, B-Subtle is also composed by:

- Metadata Collectors, which can enrich subtitles with metadata obtained from external sources, such as tools or databases;
- Producers, which can provide some additional information to the dialogue turns. A sentiment analyser, for example, can be considered a Producer;
- Transformers, which are responsible for transforming the text into dialogue turns. Lowercasing, for example, could be done by a transformer.

The functionalities provided by B-Subtle will allow us to easily select, filter, and process subtitle corpora in accordance with our needs.

4

Setup and Data Collection

Contents

We begin by presenting the datasets which were previously chosen to comprise our collection. Then, we detail the process of data retrieval which allowed us to obtain the information of our Twitter-based datasets. Lastly, we explain how we unified our dataset collection, by creating equivalences between annotation schemes and storing all of our datasets in the same file format.

4.1 Setup

Having already described the current state-of-the-art of Bias and Hate Speech Detection, as well as proposed our working definition of “Bias”, we can now focus on our developed work. As previously stated, the research question propelling our work is: ”how can pre-existing resources, namely publicly available datasets, be used to train classifiers in the task of Bias Classification – if they can be used to this end at all?”

The focus of our work results from the investigation conducted in the initial stages of development, during which we reached some of the conclusions mentioned in Section 3.1.4; namely, that the resources available in this field are not only skewed and unequal, but also that the creation of these resources is a highly costly process. Compiling available datasets and using them in a unified manner is an attractive prospect, one which would allow us to work around some of the aforementioned flaws while being less costly than, for example, building entire datasets from scratch.

The main question surrounding this approach is whether or not we can obtain a model with a good performance while using this type of training data. As we described in Chapter 3, works in Bias and Hate Speech detection are not always coherent when it concerns term usage and definition, target categories, type of training data, etc. If this lack of coherence translates to the datasets that have been developed in the past few years, then we might find they cannot be utilized conjointly in a successful manner.

In order to answer our research question, we chose publicly available datasets which differ in size (number of entries), annotation type, target categories, and data origin. Our initial selection is depicted in Table 4.1.

Dataset	Twitter-based?	Classification Type
CONAN	No	Single Target
Davidson	Yes	Binary
DynGen	No	Multi Target
Founta	Yes	Binary
Golbeck	Yes	Binary
Benevolent-Hostile Sexism	Yes	Single Target
MLMA	Yes	Multi Target
StereoSet	No	Multi Target
Waseem-Hovy	Yes	Multi Target

Table 4.1: Dataset Collection

Some of these datasets – namely, Benevolent-Hostile Sexism and Waseem-Hovy – only contain tweet identifiers, rather than the textual content of each tweet. Thus, we will have to retrieve this information before we are able to continue with our work.

4.2 Data Retrieval

In this section, we will describe the methods used to retrieve Twitter-based information from some of our datasets. We will provide a brief description of what the aforementioned process entails, as well as some of the decisions we made throughout said process. We will then examine the obtained results and the way these impacted our work.

As we have discussed in previous sections, a significant portion of our dataset collection is Twitter-based, which means that the entries of these datasets consist of content retrieved from Twitter. While Twitter is a publicly accessible platform – therefore making its content also easily accessible by the general public – there are still some privacy concerns regarding the publication of this type of dataset. Consequently, many researchers choose to not share the content of their datasets directly; rather, they share the *Tweet ID* correspondent to each entry, which can then be used to obtain its content.

A Tweet ID is an alphanumeric sequence which serves as an unique identifier to each tweet in the platform. We can easily access the content of any tweet as long as we are in possession of its identifier. Consequently, through these identifiers, we can "rebuild" a dataset like the aforementioned ones and gain access to the original information it contains.

This goal can be easily achieved by resorting to the Twitter API[1], which is available to anyone with a Developer Twitter account. The Twitter API treats each tweet as a Status object with a variety of attributes, such as the date in which it was posted, available profile information of the user who posted it, the content of the tweet itself (whether that be text, pictures, GIFs, URLs, etc.) and, of course, the Tweet ID.

When the provided Tweet ID matches an unavailable tweet, the API will not return any sort of Status object. Instead, it will raise an exception and return an error code and error message. The existent error codes and messsages are displayed in Table 4.2.

We chose to interact with the Twitter API through the Tweepy [2] Python library. Our motivation behind this choice was twofold. Firstly, since Tweepy is a Python library, it allows us to work directly with the original file format of our datasets, namely CSV and TSV, in a way that we were already familiar with. Secondly, it allows for automation of the retrieval process, which is a significant advantage when one is working with thousands of entries.

[1] https://developer.twitter.com/en/products/twitter-api
[2] https://docs.tweepy.org/en/v3.10.0/index.html

Error Code	Error Message	Cause
63	User has been suspended.	The account associated with the tweet has been suspended.
144	No status found with that ID.	The tweet or account associated with it has been deleted.
179	Sorry, you are not authorized to see this status.	The account associated with the tweet is now private.

Table 4.2: Error codes and messages of unavailable tweets

Having described the frameworks we will be working with during this phase, we can now focus on the retrieval process.

4.2.1 Tweet Retrieval

Out of our dataset collection, two datasets were made available with the Tweet IDs and without any sort of text: Benevolent-Hostile Sexism and Waseem-Hovy.

The retrieval of a tweet's content through its Tweet ID is, as previously mentioned, a basic functionality of the Twitter API. Tweepy provides us with a function that performs this task directly, which we simply have to apply to each entry of the dataset. Since we are working with CSV files, this also means that we were able to read and write directly from the files without having to resort to any auxiliary structures.

We faced two small obstacles during this task. The first was due to the way the Status object handles *retweets*. Twitter users are not only able to make their own tweets. The platform also allows users to "retweet" – that is, share directly on the platform and on their profiles – other users' tweets. Inclusively, users may insert their own commentary during the sharing process, thus adding information to the original tweet in what is called a "quote retweet", as shown in Figure 5.1.

Figure 4.1: An example tweet and quote retweet from user @ana_sevans

The way the Status object handles a retweet is by doubling the information it saves. It saves the data concerning both the original poster and the user responsible for the retweet. Consequently, it saves the content of the original tweet under the attributes text or full_text and then it saves that very same content in conjunction with any retweeted additions under the attribute retweeted_status.text or retweeted_status.full_text.

Once we realized this, our immediate choice was to prioritize the content of the retweeted_status. There were two possibilities regarding each instance of a retweet in these datasets: the first possibility is that the retweet did not add any relevant information and the content marked as biased is located in the original post; the second possibility is that the original post is innocuous, but the added information of the retweet is biased. Therefore, in this stage of our work, it is beneficial to prioritize the retweet_status.

The second obstacle we faced was that the Status object is not fully equipped to deal with Twitter's current 280 character limit. The text attribute truncates tweets with a higer character count, which is why we had to resort to full_text instead. However, retweets prove to be even a bigger challenge, because even the retweeted_status.full_text truncates retweets with a higher character count; that is to say, that while the added information of the retweet is displayed in full, the original tweet might be truncated. This presents a direct challenge to the choice we described in the previous paragraph. Therefore, in order to circumvent this problem, we decided to save the retweeted_status.full_text, which would always include the information added on retweet, and the full_text attribute, which solely saves the original tweet. Since we were working with CSV files, we decided that each attribute would be saved in its own column, and later pieced together. We will handle this issue in data treatment.

4.2.2 Interlude: Non-Persistent Data and Dataset Degradation

Let us review: due to privacy concerns, these datasets do not publicly share the textual content of the their collected tweets. Rather, we are given a Tweet ID, which we can use to retrieve the text of the correspondent tweet.

Here is the catch: we can only retrieve a tweet *if that tweet still exists*.

If we attempt to retrieve a tweet which no longer exists, or is no longer available, we will simply receive one of the error codes and messages depicted in Table 4.2. This means that some of this information is *non-recoverable* and, consequently, that Twitter-based datasets may be prone to *degradation*.

Once we realized this, we chose to not only analyse the results we had obtained in the scope of this issue, but also to repeat the retrieval process with the Founta dataset, previously presented in Section 2.2. Founta et al. [12] responded to privacy concerns by separating tweet identifiers and tweet text into separate files and then sharing both files, rather than withholding the text altogether. Ergo, while we had no need to retrieve tweets of this dataset, since the relevant information was freely provided, we still possess the identifiers and are free to use them.

The results of our analysis regarding unavailable tweets, across all four datasets, can be found in Table 4.3. The table contains the total number of tweets in the dataset, the number of available tweets, and the number of unavailable tweets, as well as why they were unavailable.

Dataset	Total Tweets	Available Tweets	Unavailable Tweets				
			Total	Suspended User	Private Account	Deleted User/Account	Other
Benevolent Sexism	7,210	2,411	4,799	1,491	375	2,925	8
Hostile Sexism	3,378	2,718	661	200	86	375	0
Founta	99,996	53,857	46,139	18,436	4,974	22,501	225
Waseem-Hovy	16,907	10,370	6,537	4,859	378	1,295	5
Total	127,491	69,356	58,136	24,986	5,813	27,096	238
Total (%)	100.00%	54.40%	45.60%	19.60%	4.56%	21.25%	0.19%
			100.00%	42.98%	10.00%	46.61%	0.41%

Table 4.3: Unavailable Tweets Breakdown

Since Benevolent-Hostile Sexism separated the Benevolent and Hostile components into two files and their yielded results differed significantly, we chose to showcase them separately.

As can be seen in Table 4.3, 45.60% of the tweets collected in these datasets had, at the time of retrieval, become unavailable. Most unavailable tweets were either deleted or posted by deleted accounts (46.61% of unavailable tweets and 21.25% of all the tweets in the datasets). A significant percentage was posted by accounts which were suspended at time of retrieval (42.98% of unavailable tweets and 10.60% of all tweets).

This is not as surprising as it might appear at first glance. On one hand, deleting an account is not an unusual phenomenon. This fact alone means that the length of time between creation of the dataset and attempted retrieval of a tweet ID contained in that dataset is proportional to the likelihood of that tweet becoming unavailable. On the other hand, and further exacerbating the previous point, Twitter allows users to flag or report content that they might find offensive. If the reported tweets are concluded to be so by Twitter's moderation team, accounts might find themselves suspended as a result. It is unsurprising that tweets belonging to a Hate Speech or Bias detection dataset might fall into this category, and thus that these datasets degrade over time.

However, unsurprising as it may be, it still warrants concern. Datasets are not only important resources, they are also inherently costly. That their value may deprecate over time due to reliance on non-persistent information presents a serious challenge, especially for a field as dependent on online-based resources as Hate Speech Detection. Perhaps solutions such as Founta et al. [12], which still address privacy concerns while circumventing the issue of degradation, should be prioritized over simply sharing Tweet IDs with little to no regard as to the preservation of the data in question.

Dataset	Twitter-based?	Classification Type
CONAN	No	Single Target
Davidson	Yes	Binary
DynGen	No	Multi Target
Founta	Yes	Binary
Golbeck	Yes	Binary
Hostile Sexism	Yes	Single Target
MLMA	Yes	Multi Target
StereoSet	No	Multi Target
Waseem-Hovy	Yes	Single Target

Table 4.4: Final Configuration of the Dataset Collection

4.2.3 Results

This dataset degradation influences the usefulness of our resources, most notably the Waseem-Hovy dataset and, in particular, the entries annotated for racism. While the original dataset boasted 1970 entries with the aforementioned label, this amount was reduced to a grand total of 12 entries. Regarding the unavailable entries, 38 entries related to deleted tweets, while 1,920 referred to suspended users.

The Benevolent Sexism portion of the Benevolent-Hostile Sexism dataset, however, yielded another problem entirely. Out of the original 7,210 tweets in total, only 2,411 remained after processing. While this may seem incredibly problematic, our main issue is actually related to the available entries. After briefly perusing the results, we realized that there seemed to be an unusual number of repeated textual content. We concluded that, out of these 2,411 available entries, only 631 were unique tweets. The remaining 1,780 entries consisted of retweets of the same original tweet, which resulted in different tweet IDs for what basically amounted to plenty of repeated content.

Both of these results had an immediate effect on our plans moving forward.

Firstly, having been reduced to a mere 631 entries, we decided to remove the Benevolent Sexism portion from our dataset collection, being left with the Hostile Sexism portion. Secondly, while we had previously considered Waseem-Hovy as a multi-target classification dataset – as a dataset which annotated entries for both the “gender” and “race” categories – the fact that only 12 entries remained for “racism” meant that this was no longer viable. Thus, we removed these entries, instead integrating the dataset into our collection as a single-target classification dataset with the target category “gender”.

The final configuration of our dataset collection can be found in Table 4.4

4.3 Data Treatment

In this section, we will expand upon the type of data processing implemented in this work. We will briefly describe the more straightforward aspects of this process, mostly related to the handling of the Twitter-based datasets, as well as expand upon some of our decisions related to the synthetic datasets. Lastly,

we will tackle the matter of label mapping.

As mentioned in the previous section, our initial collection had already suffered some changes. This was mostly a consequence of outside influences, such as our inability to reach the creators of one of our datasets, or the discovery of the dataset which we will refer to as "DynGen" [11]. The results of the retrieval process resulted in further alterations. We removed the Benevolent-Sexism portion of the Benevolent-Hostile Sexism dataset, and turned Waseem-Hovy into a "single-target classification" dataset, with the target category "Gender".

This changes, however, also meant that we were now finally in possession of all the data we had selected for our work. Thus, it was time to process it.

4.3.1 Data Processing

4.3.1.A Handling the Character Limit

As mention in Section 5.1.1, we faced a minor complication related to Twitter's 280 character limit. We decided to retrieve, and separately save, both the retweeted_status.full_text attribute, which provided us with all the information added upon retweet, as well as the full_text attribute, which contained the original tweets in full. This decision meant that we were able to bypass the Status object's inability to handle the character limit, but it also meant that we now had to piece together our information.

Retrieved retweets follow the format:

```
RT [text] @[username]:  [original tweet]
```

As a reminder, we were working with CSV files. We made the decision to save the retweet_status.full_text and full_text in different columns (RT_TEXT and TEXT respectively).

Using regular expressions, we searched every entry for the "RT" marker in the beginning of the RT_TEXT column. If found, we separated the sentence using the colon, and compared the text after the semicolon with the text contained in the TEXT column. If the two were a match, then we concluded that no truncation occurred, and saved solely the text in RT_TEXT. If these were not a match, then we knew that the text in RT_TEXT had been truncated. In this case, we added the text of the TEXT column to the text that came before the semicolon (RT [text] @[username]).

4.3.1.B Usernames, Hashtags, and Emojis

While the datasets we have been pouring over were the ones which required tweet retrieval, they were not the only datasets in our collection which were twitter-based. Davidson, Golbeck, Founta, and MLMA also fell into this category.

In this phase of our work, our main concern were text features such as usernames or hashtags, which were both extremely frequent throughout these datasets and largely irrelevant towards our classification

task. The usage of emojis might seem useful at first glance, since emojis supposedly convey feelings or reactions by supplementing textual content; however, the meaning attributed to each emoji is not universally agreed upon, and the difference in emoji usage often exposes a generational gap. Therefore, we chose to not rely on emoji usage, and treat its presence in the text in the same way we treated the aforementioned textual features.

Therefore, we chose to replace instances of these features with word markers, which would later be added to Tokenizer as special tokens. We settled on the following markers:

- All usernames were replaced with @USER. For example, the tweet *"I didn't say it, @ana_sevans did."* becomes *"I didn't say it, @USER did."*;
- All links and URLs were replaced with [URL]. The tweet *"Yet another student settles with university over an unfair process resulting from campus sexual assault hearings: http://t.co/mEWHslEByh"* becomes *"Yet another student settles with university over an unfair process resulting from campus sexual assault hearings: [URL]"*;
- All hashtags, including the hash symbol, were replaced with [HASHTAG]. The tweet *"These two are revolting #MKR #MKR2015"* becomes *"These two are revolting [HASHTAG] [HASHTAG]"*;
- All emojis were replaced with [EMOJI];
- All unicodes found in text that could not be normalized were replaced with [CHAR].

Most of these replacements - namely, urls, user handles, unicodes, and hashtags, were achieved through the use of regular expressions.

The emoji replacement was a case-by-case analysis, since it largely depended on which dataset we were processing. The datasets we obtained through the lookup process actually contained the emoji characters, while the remaining datasets had already been processed by their creators and as such were either displayed in unicode or followed some other convention. For the latter option, we were able to resort to regular expressions.

The unicode proved to be more problematic, since there is no fixed range of unicode values that can identify an emoji; namely, we cannot instruct our program to identify all unicode values as emojis as long as they are included between a pre-defined lower and upper bound. Nor, we would find out, could we decode or encode most of the values we encountered; all encoding, decoding, and normalizing functions we attempted to use and which were provided by Python failed to recognize the codes in question. Hence, we ended up resorting to a different solution, and decided to replace these instances with [CHAR].

Regarding the datasets that we obtained through lookup, we resorted to the demoji Python library[3],

[3]https://pypi.org/project/demoji/

which provides us with a replacing function which identifies all instances of emoji use in a given string and replaces them with the provided pattern – in this case, the chosen [EMOJI].

Additionally, the order in which we enforced these replacements was not trivial. Since hashtags can contain emojis, it was paramount that we could identify hashtags that did not simply contain alphanumeric characters. As such, and since, as previously mentioned, there is no convenient range of values that we can use to identify an emoji in text, we chose to firstly handle emojis and only afterwards deal with the hashtags, since it was a much simpler matter to identify the marker [EMOJI] using regular expressions.

4.3.1.C Synthetic Datasets

Once we handled all the twitter-based datasets, it was time to tackle the synthetic datasets; namely, CONAN and StereoSet. While DynGen also falls into this category, its file structure was very similar to ours, leaving us free to simply select the relevant information (mostly the text entries and the accompanying labels) and store it in a new CSV file. StereoSet and CONAN required further work.

CONAN, as described in Section 3.2, is a dataset composed of sentence pairs. Each entry features a biased sentence, which is called a *narrative*, as well as a non-biased sentence, which serves as a contradiction of the narrative - hence, it is called a counter narrative. In order to use these entries in our work, we had to separate the narrative and counter narrative pairs, as well as make sure that no narratives were repeated. This was due to the fact that while the dataset assured that no pairs were repeated, the same could not be said for the narratives or counter narratives individually. The other difficulty associated with this dataset was to select the English entries among the three languages it included, since these were not clearly separated or identified in the original files. For this, we resorted to the langdetect Python library[4], which allowed us to analyse each entry and select only the ones written in English.

Handling StereoSet was similarly a two-fold task. Firstly, the original dataset comes as a JSON file, which means we had to use the json Python library. More specifically, we resorted to the json.load() function, which generated a dictionary object from a given JSON file. Secondly, StereoSet contained two types of entries: intra-sentence, for stereotypical content which comprised a single, contained sentence; and inter-sentence, which contained a prompting unbiased sentence regarding a certain context (named, appropriately, the context), and offered three completing sentences (which we will call conclusions) - one which contained a stereotype, one which contained an anti-stereotype, and another which was fully unrelated. Since the intra-sentence and inter-sentence entries were structured differently in the dictionary and original JSON file (i.e. contained in different dictionary keys) we had to account for that difference when parsing the dictionary.

[4] https://pypi.org/project/langdetect/

Once these differences were dealt with, we selected the relevant information from both datasets and stored them in brand new CSV files.

4.3.2 Label Mapping

As previously mentioned, the datasets in our collections largely use distinct class labels. Therefore, in order to create a unified collection, we had to map a correspondence between these existing class labels and the ones we would be using going forward. This mapping process required some decision making, and followed strictly along our proposed definition of Bias.

4.3.2.A Binary Classification Mapping

The first mapping dimension we tackled was Binary Classification, i.e. simply identifying whether an entry was biased or non-biased in accordance to the definition we presented in Section 2.1.3. The correspondences detailed below are summarized in Table 4.5.

This was a straightforward process for most datasets due to an inherent biased/non-biased division. Most multi-target datasets specified this divide and used sub-labelling to define their target categories. DynGen, for example, classifies entries as “hate”/“not hate” and then specifies the target category in another column. Another example is StereoSet, which splits entries between “stereotype”, which maps to “biased”, and “counter-stereotype” and “unrelated”, both of which map to “non-biased”.

Three datasets were not as straightforward. Two of these datasets were Davidson and Founta. As described in Section 2.2, both distinguish between Hateful and Offensive content, a distinction which we have poured over in Section 2.1.2. After examining entries from both categories, we noticed that many of the entries marked as “Offensive” contained derogatory language towards our target categories, including the usage of slurs. For example:

```
Have ya ever asked your bitch for other bitches - kanye voice "Yes"
```

Is an entry from the Davidson dataset, labeled “offensive”, which clearly features gender-based slurs, namely “bitch” and “bitches”. While we concede that not all entries might follow this pattern, we decided to err on the side of caution and map both class labels (Offensive and Hateful) to biased. Additionally, after further examination, we decided to map Founta’s “spam” class to “non-biased”.

MLMA was another dataset with a non-linear correspondence. As mentioned in Section 2.1.2, the dimension we were interested in was “Level of Hostility”, which considered the following class labels: Abusive, Hateful, Offensive, Disrespectful, Fearful, and Normal. The decision process we followed towards MLMA was identical to the described in the previous paragraph, and resulted in the following split: “Abusive”, “Hateful”, “Offensive”, and “Disrespectful” mapped to biased, while “Fearful” and “Normal” mapped to non-biased. However, some entries were *multi-labeled*, that is to say, annotated with more

than one of the aforementioned labels. In these cases, and once more choosing to be cautious, we decided that every entry which featured one of the labels mapped to biased would likewise be considered biased, even if it featured the “fearful” or “normal” labels.

Hostile Sexism, as a dataset which only contained examples of gender bias, was entirely mapped to biased.

Dataset	**Binary Label Correspondence**	
	biased	**non-biased**
CONAN	hateful	normal
Davidson	hate, offensive	normal
DynGen	hate	nothate
Founta	hateful, offensive	spam, normal
Golbeck	harassment	normal
Hostile Sexism	hostile	-
MLMA	offensive, abusive, hateful, disrespectful	fearful, normal
StereoSet	stereotype	counter-stereotype, unrelated
Waseem-Hovy	sexism	none

Table 4.5: Binary Classification - Label Mapping

4.3.2.B Target Category Mapping

The second mapping dimension dealt with the target categories each dataset tackled. Due to this, Davidson, Founta, and Golbeck are not included in this section, since these these datasets solely deal with the Binary Classification task, as described in Section 2.2. The correspondences described below are summarized in Table 4.6.

In the previous section, we described that many of our multi-target datasets used sub-labels to specify the target category of each entry. We chose to apply this principle to our work. After examining our collection and thus settling on our proposed definition of “bias”, we similarly decided on the following class labels: gender, race, profession, religion, disability, sexual_orientation, gender_identity, nationality, b_none, and non-biased. While most of these are self-explanatory, we would like to expand upon the last two. non-biased is the sub-label correspondent to the non-biased label we have previously presented. b_none refers to entries which are annotated as biased, but either do not specify a target (like the binary classification datasets we have already mentioned) or are annotated in their original datasets as “other”.

Once more, the Target Category Mapping was a straightforward process for most datasets. Hostile Sexism, Waseem-Hovy and CONAN only targeted one well established category (i.e. gender for the first two and religion for the latter). Others, such as StereoSet and MLMA, used terminology which was similar or identical to ours.

The exception was DynGen. DynGen not only utilizes an intersectional labelling approach (for example, bla.wom for “black women”, gay.man for “gay men”, mus.wom for “muslim women”), as it also uses

highly specific labels (asi, asi.east and asi.south are all distinct labels). Additionally, some of its entries are also annotated with more than one label. Thus, we had to deal with each of these issues separately.

We started out by separating the labels which referred to a single identity and directly corresponded to our existent categories. For example, “bla” and “non.white” could be directly mapped to race, “wom” (women) to gender, “russian” to nationality, etc. Then, we had to separate those labels which, while clearly referring to a single identity, could theoretically fit into more than one of our categories. For example, we made a broad decision that any label which referred to a specific country would map to nationality, while general area identifiers would map to race (for example, “chinese” in opposition to “asi”, “asi.east”, “asi.south”).

Afterwards, we decided to tackle the intersectional labels. We observed that most of these labels were composed by a gender identifier and an identifier of another category (“bla.wom”, “asi.wom”, “mus.wom”, “gay.wom”). Since we already had plenty of entries with the class label gender, we decided that, *in most cases*, these labels would map to the category of the non-gender identifier (for example, “bla.wom” would map to race).

Once this was settled, we were left to deal with the entries annotated with more than one label. In this case, we followed a “Majority Rules” approach; namely, if most or all class labels of a certain entry mapped to one of our categories, then the entry itself would as well (ex: “bla.wom, bla, dis” would map to race). This is why, in the previous paragraph, we emphasized that the described decision could be overruled; for example, if an entry is annotated with “wom, gay.wom”, “Majority Rules” is applied and the entry is labeled as gender. If “Majority Rules” was not applicable, which happened in some cases, we resolved the mapping of those cases by hand, on a case-by-case basis.

Category	DynGen	MLMA	StereoSet
gender	wom	gender	gender
race	mixed.race, ethnic.minority, indig, indig.wom, non.white, non.white.wom, trav, bla, bla.wom, bla.man, african, asi, asi.man, asi.wom, asi.south, asi.east, arab, immig, asylum, ref, for, hispanic, nazis, hitler	origin	race
profession	wc, working	-	profession
religion	jew, mus, mus.wom, other.religion	religion	religion
disability	dis	disability	-
sexual_orientation	bis, gay, gay.man, gay.wom, lgbtq	-	-
gender_identity	trans, gendermin	-	-
age	old.people		
nationality	eastern.europe, russian, pol, chinese, pak, asi.chin, asi.pak, other.national	-	-
b_none	none, notgiven, notargetrecorded, NA, other.glorification	other	-

Table 4.6: Multi-Target Classification - Label Mapping

Once these issues were dealt with, the label mapping process was finished. Our dataset collection was finally unified and ready to be used.

5

Model Training

Contents

Once we were in possession of our prepared dataset collection, it was now time to utilize it. In this chapter, we describe the Experimental Setup we utilized in our work, as well as the results obtained from our experiments. Furthermore, we analyse those results in the scope of the research questions ascribed to each data control group in Chapter 4, and derive conclusions that inform the rest of our work.

5.1 Experimental Setup

5.1.1 The Model

The goal of this work is fundamentally exploratory and primarily focused on taking advantage of pre-existing resources. Consequently, it is out of the scope of this work to develop new model architectures or, even, to implement a model of our own at all. Therefore, we chose to simply utilize a previously existing model and alter it to better accommodate our needs.

With this in mind, we chose the Emotion-Transformer [1]. The Emotion-Transformer was developed in the scope of Emotion Detection, as the name might suggest, but it is easily adaptable to our Bias Classification task. The only significant change to the original code was the addition of special tokens to the Tokenizer – namely, to handle the [EMOJI], [CHAR], [HASHTAG], @USER, and [URL] markers, mentioned in Section 4.3.1.B.

Furthermore, the Emotion-Transformer is built on top of a pretrained Transformer model. After some consideration, we decided to use the DistilBERT pretrained model from HuggingFace for our classification task [2]. This choice was largely motivated by temporal constraints; while models such as RoBERTa may perform much better than DistilBERT, this performance comes with a significant temporal cost. After examining the number of experiences we planned to complete, as well as how the available hardware behaved in relation to the different pretrained models, we concluded that DistilBERT was a necessary compromise between temporal efficiency and overall performance. Nevertheless, it bears acknowledging that other pretrained models could yield much better results, if given the chance. We shall leave that particular experiment to future work.

In order to establish the Emotion-Transformer's level of performance, we decided to train it with the individual datasets of our collection and compare the obtained results against results reported in the publication of those very same datasets. We concluded that any comparison of results for Benevolent-Hostile Sexism and Waseem-Hovy would be invalid, due to the alterations these datasets suffered (described in Chapter 4). Out of the remaining datasets, only Davidson, DynGen, and MLMA reported performance results. Additionally, DynGen was evaluated in a multi-labeling task, which would make our evaluation of it as a single-labeling task fully irrelevant, since we would essentially be comparing different tasks. Thus,

[1] https://github.com/HLT-MAIA/Emotion-Transformer
[2] https://huggingface.co/docs/transformers/model_doc/distilbert

we performed the train/validation/testing split on both Davidson and MLMA, and separately trained and tested the Emotion-Transformer with both datasets.

Davidson originally reported an F1-score of 0.9, using a Support Vector Machine with L2 regularization [37]. MLMA does not specify what type of methods were used in training and testing, instead presenting a range of results for the different dimensions of the dataset. As mentioned in Section 2.2, the dimension relevant to our work is *TARGET*, which reported an F1-score 0.43 as its best result.

In our testing, we were able to obtain an F1-score of 0.8 for Davidson, training the Emotion-Transformer during 5 epochs, with Binary Cross-Entropy with Logits Loss and max pooling function; and an F1-score of 0.42 for MLMA, training the Emotion-Transformer during 6 epochs, with the same Loss and Pooling functions described for the previous experiments. While the F1-score obtained for Davidson is slightly lower than originally reported, the values are still quite similar. Thus, we conclude that the Emotion-Transformer is able to perform at a similar level to those models used to test the original datasets.

Having described the model we used, and established its level of performance, we will now detail the experiments conducted in this phase of our work.

5.1.2 The Experiments

We began Chapter 4 by presenting the datasets which were previously selected to be part of our dataset collection. Later on in that same chapter, we revised our selection slightly, due to issues with the tweet retrieval process. Namely, we removed the Benevolent Sexism portion of the Benevolent-Hostile Sexism dataset, thus dealing only with the Hostile Sexism portion going forwards, and also adopted Waseem-Hovy as a Single-Target Classification dataset rather than a Multi-Target Classification dataset, which was its initial designation. The final configuration of our the dataset collection can be found in Table 4.4.

In order to properly answer our research question, first presented in Chapter 1 and then revisited in Chapter 4, we decided to separate our dataset collection into four non-exclusive groups, named Group A, Group B, Group C, and Group D. Group A, as the smallest, most coherent, and least complex of the groups, serves as our baseline for performance comparison. Each of the three remaining groups is meant to answer a particular research question, and serves a different exploratory focus. These questions, as well as the composition of each control group, is described in Table 5.1.

The goal of this work phase is twofold. Firstly, we mean to answer the research questions posed and described in Table 5.1, which largely focus on how the several individual datasets perform when utilized as a single, coherent resource. Secondly, we mean to find the control group and model parameters that overall perform better in testing, in order to use them in the next phase of our work. This allows us to understand if our best-performing model, trained with a given, pre-existing dataset collection, is able to accurately classify biased content in training data meant for specific downstream tasks.

In order to achieve this second goal, we had to conduct several experiments. We performed a non-

Group Name	Datasets	Questions
A	Davidson + Founta + Golbeck	Baseline
B	Group A + Hostile Sexism + Waseem-Hovy	How do single-target datasets influence performance?
C	Group A + DynGen + MLMA + StereoSet	How do synthetic and multi target datasets influence performance?
D	Group C + CONAN + Hostile Sexism + Waseem-Hovy	Can we obtain better performance by using all of our resources together?

Table 5.1: Dataset Groups

deterministic split of each group's data, splitting it into training, testing, and validation sets (80% train and 10% for testing and validation each) using scikit-learn [3].

Since Group A was the smallest group by far, we used the models trained on Group A data to figure out which model parameters had the biggest impact on the model's overall performance, and which experiments were worth replicating with the other dataset groups. We conducted a total of 67 experiments with Group A, in which we studied the effect of different pooling functions, loss functions, seed value, learning rate of the classification head, number of frozen epochs, and number of total epochs.

Out of these, we concluded that the most influential parameters were the Pooling Function (namely, those implemented in the Emotion-Transformer: avg, max, and cls), Loss Function (Binary Cross Entropy with Logits Loss, L1Loss, and Cross Entropy), and the Total Number of Epochs. Therefore, these were the variables we tested in the experiments regarding Group B, C, and D.

The rest of the parameters were not altered for the remaining tests. The values for each parameter were those defined as default in the Emotion-Transformer, and are as follows:

- **Seed Value**, which was set at 12
- **Patience**, which was set at 1 and refers to the number of epochs allowed to run without noticeable improvement before training stops
- **Gradient Accumulation Steps**, which was set at 1
- **Batch size**, set at 8
- **Number of Frozen Epochs** was set at 1
- **Encoder Learning Rate**, set at 1.0e-5
- **Classification Head Learning Rate**, set at 5.0e-5
- **Layerwise Decay**, set at 0.95
- **Deterministic Training**, set at True, thus ensuring that our experiments could be replicated.

[3] https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.train_test_split.html

5.1.3 Interlude: Class Imbalance, Undersampling, and Data Augmentation

As we considered and explained the intricacies of our work, there is an aspect that we continuously emphasized: namely, the unequal distribution of resources regarding Bias and Hate Speech Detection, as well as the consequences that imbalance has on our own work.

The questions that we seek to answer in this phase of our work – namely, those detailed in Table 5.1 – are largely derived from this concern. How do single-target datasets influence performance? Or, in other words, *can* they influence performance at all? Does the addition of single-target training data, even if in a lower quantity than the binary-classification data, detract from the baseline performance? Or does it make the model better, and if so, in what way? How do synthetic and multi target datasets influence performance? Can the model adapt to the different linguistic conventions of synthetic and online-based data? Are the small amounts of training data regarding a particular protected category *enough*?

These questions might not have been necessary, had we been working with copious amounts of well-balanced data. However, that is surely not the case. As we mentioned in Section 3.3, one of the most blatant limitations of this field of study, at the moment, is the way certain target categories (most notably, "Gender" and "Race") receive a lot more attention – and, as such, a lot more dedicated resources – than any other category. This skewed distribution has had an obvious impact in our work; not only is our single-target control group focused on the target category "Gender", but also the distribution of available resources across our chosen target categories is glaringly skewed, as can be seen in Table 5.2 and Table 5.3. Table 5.2 details the split between biased and non-biased entries in each dataset, while Table 5.3 splits biased content into target categories.

	Non-Biased	**Biased**	**Total**
Group A	81,112	44,016	125,128
Group B	88,754	49,449	138,203
Group C	109,265	75,341	184,606
Group D	120,851	81,289	202,140

Table 5.2: Breakdown of Biased and Non-Biased Entries, represented by number of entries per label

Group A features a fairly balanced data split, with the majority class loosely making up two thirds of the total available data. Although this is not evenly split data, is still a fairly balanced dataset, especially once one considers the sheer number of entries. The distribution of target categories across other groups, however, is blatantly skewed.

The question arises: is it possible to somehow balance this data ourselves? Well, in theory, yes. In practice, it is more complex than that.

One way of balancing a previously imbalanced dataset is through Undersampling; namely, removing entries from majority classes until we are close to an even split across classes. While this is, at first glance, a seemingly reasonable solution, it is a solution that we simply cannot implement in our work.

	Group B	B (%)	Group C	C (%)	Group D	D (%)
Non-Biased	88,754	64.22%	109,265	59.19%	120,851	59.79%
Biased (None)	44,016	31.85%	51,947	28.14%	51,947	25.70%
Gender	5,433	3.93%	3,182	1.72%	8,615	4.26%
Race			10,613	5.75%	10,613	5.25%
Profession			1,855	1.00%	1,855	0.92%
Religion			2,632	1.43%	3,147	1.56%
Disability			1,575	0.85%	1,575	0.78%
Sexual Orientation			1,854	1.00%	1,854	0.92%
Gender Identity			1,132	0.61%	1,132	0.56%
Nationality			528	0.29%	528	0.26%
Age			23	0.01%	23	0.01%

Table 5.3: Breakdown of Entries of each target category, represented by number of entries per label

Particularly when it comes to Groups C and D, some of the classes do not reach a thousand entries. "Age", in particular, features 23 entries in total. Undersampling would, quite simply, sabotage our performance by heavily reducing the amount of available data to a meager portion, which would not be enough for our model to learn from. Therefore, Undersampling, while a noteworthy technique in itself, is not suited for our work.

The other way of balancing a dataset is by turning to the opposite direction: if we cannot remove entries, then we shall add new ones. The most direct way to achieve this is also one which is fully unavailable to us; namely, collecting new data, annotating it in accordance to our Bias definition and target categories, and simply add it to the existing data. As previously mentioned, this is a highly costly and time-consuming process; it is, in fact, costly and time-consuming enough that it motivates and emphasizes our focus on working with previously existing data.

Another common way of augmenting a dataset is Data Augmentation, which is the process of creating new data by altering copies of pre-existing data. This alternative is much more accessible than creating brand new data, and while it does have its setbacks – namely, it can result in a stale and/or repetitive dataset, if used in excess – it seems, at first glance, to be an excellent solution to our problem.

This is, however, not the case. As we have also repeatedly mentioned, one of the most complicated aspects of this field of study is the fact that "bias" is not a fixed category, with unanimously agreed-upon manifestations. Certainly, there are some instances in which simply grabbing a biased sentence and replacing a word related to a target category by a word related to a different category would successfully result in a brand new biased sentence. For example, if one were to look at the sentence "I hate Muslims!", and simply swap "Muslims" for, say, "Nurses", we would find ourselves with a brand new sentence which exhibited negative sentiment regarding the target category "Profession".

However, the types of biased entries in our datasets – and the way bias often manifests in real life – are often not this straightforward. We often consider certain sentiments or sentences to be biased not because of their inherent nature, but because they refer to a target category in a way that, in our

sociocultural framework, is considered biased. “All girls are terrorists.” and “All Muslims are terrorists.” are both sentences which contain a generalization; however, only the second sentence represents a *stereotype* – or, in other words, “a preconceived notion” of a group of people which, quite often, results in unequal treatment of individuals perceived to be part of that very same group. *This* is our definition of bias; not just any type of generalization.

Bias and Hate Speech are not concepts which exist in a vacuum, and can be carelessly replicated by simply swapping word pairs. We cannot divorce these concepts from the realities they represent without robbing them of their inherent meaning and fundamentally changing the aim of our work. Therefore, while we have had to make a number of concessions regarding the way we explore bias in our work – namely, due to our inability to conduct our research through an Intersectional lens (mentioned in Section 2.2) and due to the modifications of the datasets in our collection, described in Section 4.3.2 – we decided that applying Data Augmentation, while possibly lessening the impact of our imbalanced classes, would also distance us from the work we truly wanted to do.

Hence, we decided to continue working with imbalanced datasets, shifting our exploratory focus to also analyse how this imbalance would impact model performance. We invite future work to further explore the possibility of balancing these types of datasets, and how to achieve that goal without compromising the complexity of the phenomenon being studied.

5.2 Initial Results

In this section, we will address and describe the most relevant experiments conducted during this work phase. This mostly refers to experiments which tested the relevant parameters described in the previous section, or experiments which proved useful to our analysis and discussion of the research questions posed earlier in this chapter.

5.2.1 Baseline Performance: Group A vs Groups B, C, and D

As previously mentioned, Group A will serve as the baseline against which we will be comparing the performance of the models trained with the remaining groups. However, Group A is also the only group that can only be used for the Binary Classification task. The other three groups can be used in this way if we instruct the model to solely consider the binary biased/non-biased labels in the LABEL column; or, they can be used to train a model to also identify the target category of a biased sentence, by considering the category labels in the SUB_LABEL column of each file. In other words, Groups B, C, and D can be used in both Binary Classification *and* in Multi-Target Classification.

Therefore, in other to obtain proper comparison against our baseline, we conducted three types of tests. The first type was in Multi-Target Classification, using Groups B, C, and D, in which both the

training and testing data were from the same group. The second type was in Binary Classification, using all groups, in which both the training and testing data were also from the same group. The third type was also in Binary Classification – but we used a Model trained with data from Group A to classify test data from Groups B, C, and D.

Since Group A remains unchanged across the different types of tests, we will continue to refer to experiments run with Group A simply as "Group A".

The F1-scores for the first type of test are depicted in Table 5.4, Table 5.5, and Table 5.6. Each depicts the three best results for each experiment (E1, E2, and E3, respectively), with the best overall F1-score result in bold, and the parameters applied in each case. We will refer to the groups of experiments in this type of the test as "Multi-B", "Multi-C", and "Multi-D". Additionally, we will refer to the Binary Cross-Entropy with Logits Loss Function as simply "BCE". Lastly, due to the high number of categories in Multi-C and Multi-D, we will preserve the readability of the present document by simply presenting the F1-scores in Table 5.6. The full value spread, which includes Precision and Recall for all categories, will be included in Appendix A.

The experiments with Group A yielded interesting results. Models trained with BCE for 3 to 7 epochs, inclusively, produced the exact same Precision, Recall, and F1-score values in testing, differing only according to the Pooling Function applied. This phenomenon did not occur during the remaining experiments and happened consistently once we tried to replicate the experiment. Due to this, we have chosen to circumvent this redundancy, and represent the number of epochs during which the same value was observed. Group A, as expected, yields the best overall results, as shown in Table 5.4. Precision and Recall are very similar for both labels, and although the model performs better with the non-biased category, the difference is not significant.

In Multi-B's case, depicted in Table 5.5, we consider the unspecified "biased" category as well as the target category "Gender". As previously mentioned, the data labeled for this category was obtained from Twitter-based datasets, similarly to the datasets which compose our baseline. Multi-B shows a balanced performance for Precision and Recall across categories, as well as a slight decrease in performance when compared to the baseline results.

Multi-C and Multi-D, both depicted in Table 5.6, are a completely different story. They both show a severe decrease in performance when compared to both Group A and Multi-B, and the F1-scores for each category fluctuate widely. Unsurprisingly, the model performs better for the "Non-Biased" (non-b) and unspecified "biased" (b_none) categories, both of which had a significant number of entries. The remaining categories, however, do not perform quite so linearly; the "Gender Identity" (gen_id) category, for example, which represented 0.61% and 0.56% of Groups C and D, respectively, shows best overall performance than, for example, "Race", which represented 5.75% and 5.25% of all data from Groups C and D (information detailed in Table 5.3).

	Epochs	Pooling	Loss	biased			non-biased			Overall		
				P	R	F1	P	R	F1	P	R	F1
A-E1	3-7	avg	BCE	0.8527	0.8782	0.8653	0.9364	0.9229	0.9296	0.8946	0.9006	**0.8974**
A-E2	8	avg	BCE	0.8502	0.8789	0.8643	0.9377	0.9212	0.9294	0.8940	0.9000	0.8968
A-E3	3-7	max	BCE	0.8574	0.8717	0.8645	0.9319	0.9248	0.9283	0.8946	0.8982	0.8964

Table 5.4: Group A: Best Results

	Epochs	Pooling	Loss	biased			gender			non-biased			Overall		
				P	R	F1	P	R	F1	P	R	F1	P	R	F1
B-E1	6	avg	BCE	0.8286	0.8906	0.8585	0.8736	0.8637	0.8686	0.9398	0.9116	0.9255	0.8806	0.8886	**0.8842**
B-E2	4	avg	BCE	0.8530	0.8741	0.8634	0.8292	0.8677	0.8480	0.9296	0.9187	0.9241	0.8706	0.8868	0.8785
B-E3	4	cls	BCE	0.8530	0.8731	0.8629	0.8588	0.8359	0.8472	0.9280	0.9201	0.9240	0.8799	0.8764	0.8780

Table 5.5: Multi-B: Best Results

	Epochs	Pooling	Loss	b_none	gender	race	prof.	rel.	dis.	sex_or	gen_id	nation.	age	non-b	overall
C-E1	6	max	BCE	0.8425	0.5875	0.4621	0.4257	0.4243	0.5128	0.6304	0.6038	0.4910	0	0.8796	**0.6046**
C-E2	6	avg	BCE	0.8456	0.5969	0.1124	0.6254	0.6296	0.6758	0.6190	0.5960	0.5000	0	0.8814	0.6020
C-E3	4	max	BCE	0.8449	0.5519	0.6392	0.6355	0.5937	0.6364	0.6258	0.6011	0.5143	0	0.8863	0.5935
D-E1	4	avg	BCE	0.8377	0.7458	0.6619	0.6499	0.6598	0.6245	0.6527	0.5424	0.4938	0	0.8770	**0.6132**
D-E2	6	max	BCE	0.8382	0.7453	0.6461	0.6217	0.6591	0.6345	0.6504	0.5310	0.5236	0	0.8801	0.6117
D-E3	4	max	BCE	0.8372	0.7432	0.6613	0.6452	0.6501	0.6224	0.6460	0.5553	0.4841	0	0.8768	0.6112

Table 5.6: Multi-C and Multi-D: Best Results

Further discussion on these results shall be conducted in Section 5.2.2.

The second type of test was conducted across 12 experiments, evenly split across Groups B, C, and D. The results are depicted in Tables 5.7, 5.8, and 5.9. We will refer to the groups of experiments in this type of test as “Binary-B”, “Binary-C”, and “Binary-D”. The experiments themselves will be identified by a number, preceded by their group’s identifying letter. The variation between experiments was in the Number of Epochs (4 (E4 and E5) or 6 (E6 and E7)) and in the Pooling Function (avg (even numbers) and max (odd numbers)). All experiments used the BCE Loss Function.

	biased			**non-biased**			**Overall**		
	P	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
B-E4	0.8249	0.8934	0.8578	0.9441	0.9047	0.9240	0.8845	0.8991	**0.8909**
B-E5	0.8301	0.8871	0.8577	0.9389	0.9080	0.9232	0.8845	0.8976	0.8904
B-E6	0.8248	0.8933	0.8577	0.9435	0.9045	0.9236	0.8841	0.8989	0.8906
B-E7	0.8304	0.8879	0.8582	0.9228	0.9069	0.9392	0.8848	0.8974	0.8905

Table 5.7: Binary-B Results Breakdown

	biased			**non-biased**			**Overall**		
	P	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
C-E4	0.8147	0.8488	0.8314	0.9004	0.8760	0.8880	0.8576	0.8624	**0.8597**
C-E5	0.8163	0.8476	0.8317	0.8971	0.8772	0.8871	0.8567	0.8624	0.8594
C-E6	0.8147	0.8488	0.8314	0.9004	0.8760	0.8880	0.8576	0.8624	**0.8597**
C-E7	0.8163	0.8476	0.8317	0.8971	0.8772	0.8871	0.8567	0.8624	0.8594

Table 5.8: Binary-C Results Breakdown

	biased			**non-biased**			**Overall**		
	P	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
D-E4	0.8039	0.8366	0.8199	0.8938	0.8724	0.8830	0.8489	0.8545	**0,8515**
D-E5	0.8238	0.8145	0.8191	0.8738	0.8817	0.8777	0.8488	0.8481	0.8484
D-E6	0.8259	0.8141	0.8200	0.8741	0.8827	0.8784	0.8500	0.8484	0.8492
D-E7	0.8238	0.8145	0.8191	0.8738	0.8817	0.8777	0.8488	0.8481	0.8484

Table 5.9: Binary-D Results Breakdown

	biased			**non-biased**			**Overall**		
	P	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
Inter-B	0.7876	0.8974	0.8389	0.9489	0.8871	0.9170	0.8683	0.8923	**0.8780**
Inter-C	0.6563	0.8152	0.7272	0.8978	0.7908	0.8409	0.7770	0.8030	0.7840
Inter-D	0.6150	0.8027	0.6964	0.8989	0.7772	0.8336	0.7570	0.7899	0.7650

Table 5.10: Intergroup Results Breakdown

The results depicted in Tables 5.7, 5.8, and 5.9 follow a similar pattern to Group A, in Table 5.4; namely, similar values for Precision and Recall, slightly better performance for the “Non-Biased” class, and very consistent overall results. Further discussion of these results in the following section.

The third type of test was conducted using the best performing model trained with Group A data.

This model was trained in 4 epochs, with the avg Pooling Function and BCE Loss Function. The results are depicted in Table 5.10. We will refer to these experiments as “Inter-B”, “Inter-C”, and “Inter-D”.

5.2.2 Interlude: The “Age” Category Conundrum

In an unsurprising turn of events, both the C and D groups yielded a null value for Precision, Recall, and F1 for the “Age” category (information depicted in Appendix A). This is unsurprising due to the extremely low number of entries for this category, which, in both groups, amounts to a grand total of 0.01% of all entries (as shown in Table 5.3). Not only is this not enough to properly train the model, as the data split between train, validation, and test also ensures that very few entries make it into the testing phase to begin with.

Once this problem became apparent, the next step was clear: we would remove all 23 entries labeled age from groups C and D and retrain the model in the task of Multi-Target Classification. Then, we would compare and analyse the obtained results, not only in terms of overall performance, but also in the way this change altered the model’s ability to correctly identify instances of the other target categories. We will refer to these experiment groups as “NoAge-C” and “NoAge-D”.

F1-score values of the several target categories are depicted in Table 5.11 and Table 5.12. Once more, we will include the full value spread in Appendix A. The variation between experiments was in the Number of Epochs (4 and 6) and in the Pooling Function (avg and max). All experiments used BCE Loss Function. The experiments themselves as will be identified by their group letter and a number from 8 to 11. Experiments numbered 8 and 9 were trained during 4 epochs, while 10 and 11 were trained during 6 epochs. Furthermore, even numbers used the avg Pooling function while odd numbers used the max Pooling function.

	b_none	**gender**	**race**	**prof**	**rel.**	**dis.**	**sex_or**	**gen_id**	**nation.**	**non-b**	**Overall**
C-E8	0.8443	0.5626	0.6235	0.6411	0.6147	0.6328	0.5849	0.5495	0.4000	0.8876	0.6341
C-E9	0.8414	0.5669	0.6254	0.6358	0.6240	0.6468	0.6059	0.5851	0.4528	0.8859	0.6470
C-E10	0.8436	0.6030	0.6452	0.6551	0.6586	0.6825	0.6199	0.4276	0.5229	0.8861	0.6544
C-E11	0.8475	0.6054	0.6545	0.6494	0.6569	0.6935	0.6309	0.5756	0.5714	0.8847	**0.6770**

Table 5.11: NoAge-C: Breakdown of F1-score results

	b_none	**gender**	**race**	**prof**	**rel.**	**dis.**	**sex_or**	**gen_id**	**nation.**	**non-b**	**Overall**
D-E8	0.8359	0.7346	0.6540	0.6055	0.6359	0.6331	0.6047	0.5053	0.5238	0.8861	0.6619
D-E9	0.8379	0.7313	0.6587	0.6220	0.6331	0.6394	0.6280	0.5563	0.5455	0.8864	0.6738
D-E10	0.8327	0.7402	0.6532	0.6359	0.6471	0.6443	0.6571	0.5345	0.5063	0.8766	0.6728
D-E11	0.8307	0.7439	0.6380	0.6376	0.6532	0.6419	0.6636	0.5438	0.5698	0.8774	**0.6800**

Table 5.12: NoAge-D: Breakdown of F1-score results

In order to obtain a valid comparison, we decided to compare the NoAge-C and NoAge-D experiments with their Multi-C and Multi-D counterparts trained with the same parameters, i.e., either 4 or 6

epochs, avg or max Pooling Function, and Binary Cross Entropy with Logits Loss. Figure 5.1 and Figure 5.2 show the F1-scores of the Multi-C and NoAge-C experiments with 4 and 6 epochs, respectively. Figure 5.3 and Figure 5.4 fulfill the same purpose for Multi-D and NoAge-D. Lastly, Figure 5.5 depicts the average F1-score from NoAge-C and NoAge-D, obtained from the four experiments conducted, as well as the average F1-score from the Multi-C and Multi-D counterparts trained with the same parameters.

We can observe from Figure 5.5 that the overall F1-score of the experiments increased after removing the "Age" category, which makes sense since no longer are any null scores to drag the overall score down. Both the b_none and non-biased labels remain unchanged, each representing over 20% and 50%, respectively, of groups C and D. This is unsurprising, due to the fact that entries labeled "Age" represented 0.01% of either dataset. It stands to reason that the removal of few entries would not cause a significant change to the biggest classes.

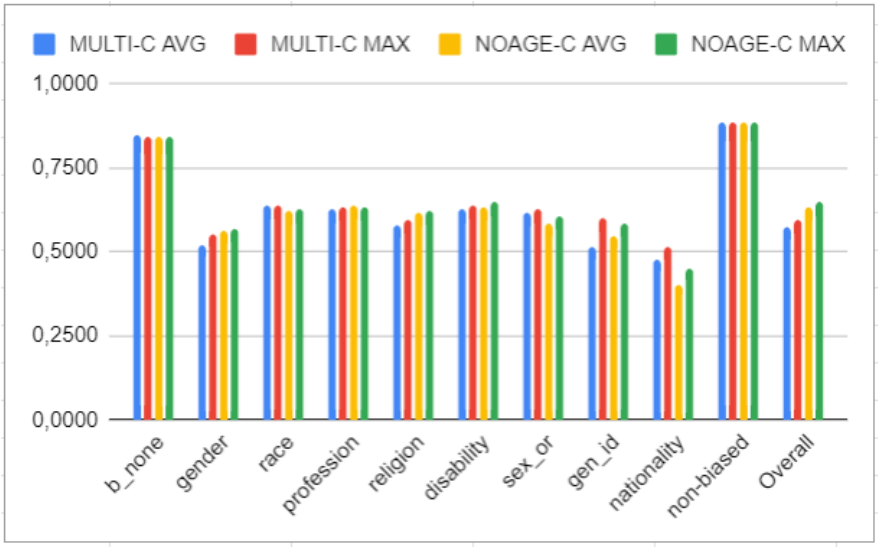

Figure 5.1: F1-scores of Multi-C and NoAge-C experiments trained during 4 epochs

We can also, however, clearly observe variations equal to, or over, 0,03 between the average F1-scores. For Multi-C and NoAge-C, this variation can be observed in race (5.75% of Group C), religion (1.43%), disability (0.85%), and gender identity (0.61%). For Multi-D and NoAge-D, we only observe a variation of this magnitude in nationality (0.26%).

When it comes to classes that represent a smaller percentage, such as disability, gender identity, and nationality, it makes sense that even small changes in the dataset could result in changes in the model's performance. Since the model has less data to learn from, the removal or addition of entries or classes is more easily noticed in smaller classes. Furthermore, the fact that some of these categories suffered variations in one Group and not the other can be easily explained by chance; a different split between train, validation, and testing, or perhaps a different seed value, could result in these variations happening to other classes, in different iterations of these experiments. Even the results observed for

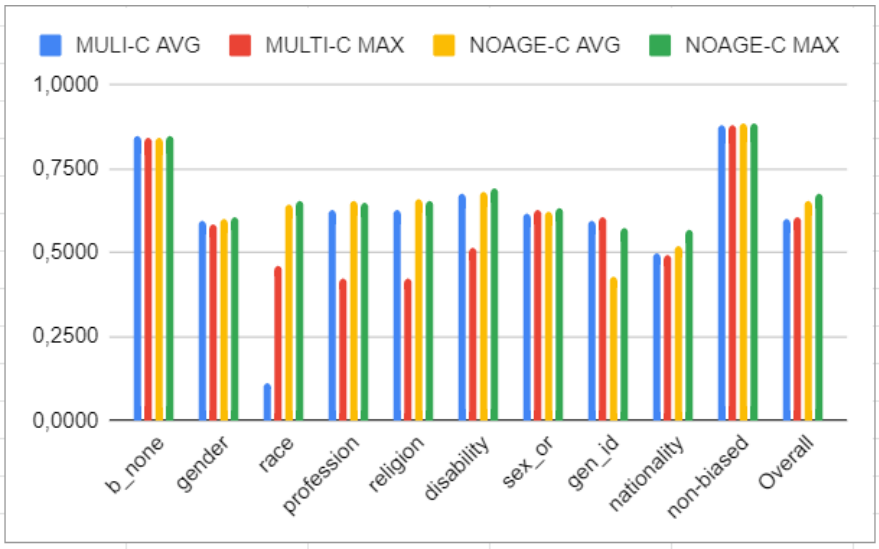

Figure 5.2: F1-scores of Multi-C and NoAge-C experiments trained during 6 epochs

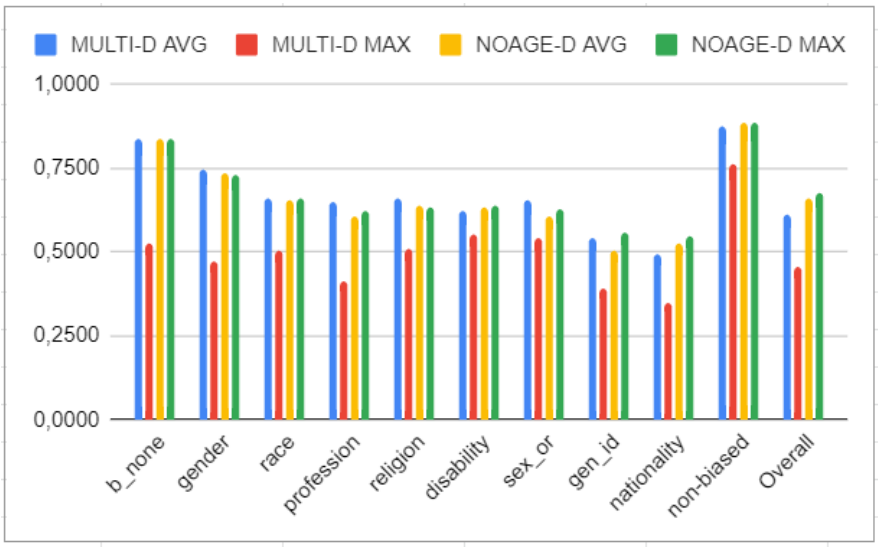

Figure 5.3: F1-scores of Multi-D and NoAge-D experiments trained during 4 epochs

religion, which represents 1.56% in Group D, can be supported by this hypothesis.

What does not factor into this hypothesis is the variation observed in race in Group C, which is also the biggest variation observed in Figure 5.5 (values differ in 0,09). Thankfully, the observed discrepancy is an anomaly, originated by a different, unrelated anomalous result. As stated, the values depicted in Figure 5.5 are F1-score averages from a number of experiments, some of which are depicted in the tables presented in the previous subsection. Namely, experiment C-E2, which can be found in Table 5.6, has a very low F1-score on race. This is due to an extremely low Recall score (0.0596, to be precise) and a high Precision score (0.9787). This information is detailed in Appendix A.

Therefore, the large difference between average scores for race is not related to the “Age” conun-

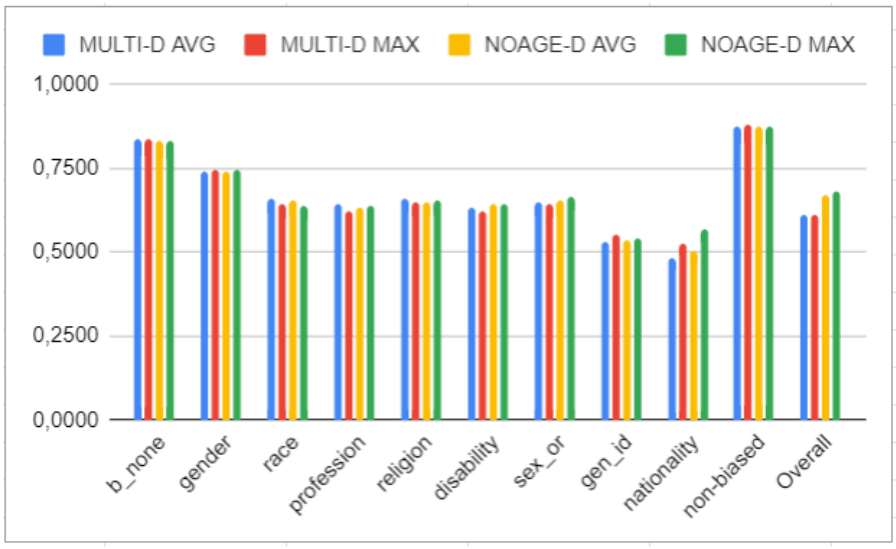

Figure 5.4: F1-scores of Multi-D and NoAge-D experiments trained during 6 epochs

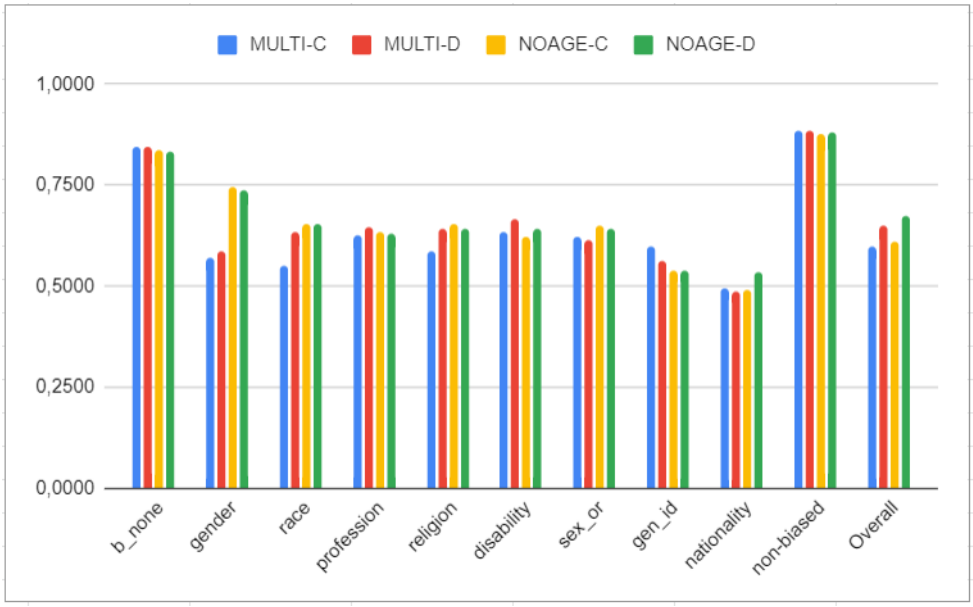

Figure 5.5: Average F-scores of Multi-C, Multi-D, NoAge-C, and NoAge-D

drum. Our hypothesis here is that the affected classes are the smaller classes, which are more vulnerable to changes in the training data. Furthermore, the fact that these shifts in behaviour are not uniform in both sets of experiments (namely, only being observed in Multi-C/NoAge-C or in Multi-D/NoAge-D, never in both), is simply due to chance. The overall score of the system increases with the removal of the “Age” category not because of any real improvement in performance, but simply due to the way the score is calculated.

5.3 Answering the Dataset Group Questions

5.3.1 "How do Single-Target datasets influence performance?" Or: Group-A vs Multi-B, Binary-B, and Inter-B

This is the question that led us to create Group B as a distinct control group, with its sole Target Category. Furthermore, since all the individual datasets in this group are Twitter-based, we also remove other variables from this experiment, such as the linguistic variation of Internet and synthetic data.

As can be seen in Table 5.4 and Table 5.5, the difference in overall performance between Group A and Multi-B is slight. We observe a 0.01 to 0.02 decrease in F1-score overall, but neither the non-biased nor the unspecified biased category see any decrease in performance. gender and biased yield very similar and high scores. From this, we can conclude that the model is able to correctly predict when a sentence is biased, and also when that bias is aimed at target category gender.

Observing the Binary-B results, shown in Table 5.7, we can see a 0.01 decrease in F1-score in the biased category when compared to Group A's results. While the model's ability to differentiate between biased and non-biased content is maintained, we can presume that the entries from the Single-Target datasets differ enough from the unspecified biased entries to result in a noticeable, yet slight, decrease in performance. This addition does not seem to impact the non-biased category in any significant way.

Lastly, let us observe the results obtained in Inter-B, and compare them to those experiments in Group-A and Binary-B which used the same parameters, namely, A-E1 and B-E4. Inter-B's F1-score of 0.8780 compared to A-E1's 0.8974 shows us that the model solely trained on Group A data, while clearly able to identify *some* of the gender-biased entries and perform rather adequately, does not perform as well as the baseline. Most importantly, it also does not perform as well as a model trained with Group-B data, as evidenced by B-E4's F1-score of 0.8909.

We can conclude that adding entries labeled for a specific target category to a general Bias/Hate Speech dataset results in a model which can accurately identify and classify biased content revolving around that very same target category, with little to no decrease in overall performance. While the difference between the obtained values is relatively minor, it becomes all the more impressive once one remembers that gender-annotated entries represent a mere 3.93% of the total data available in Group B (Table 5.3). These results are, therefore, highly promising.

5.3.2 "How do synthetic and Multi-Target datasets influence performance?" Or: A Lukewarm Overview of Group C

This is the question that motivated the existence of Group C as a control group, by adding to our baseline those datasets that were Multi-Target and/or synthetic. This was an almost by default choice, since most

of our Multi-Target datasets were also synthetic.

As depicted in Table 5.6, the difference in performance between Group A and Multi-C is significant, even with the increase observe by removing the "Age" category, as depicted in Table 5.11. Group A's overall F1-score rests solidly in the 0.89 range, while Multi-C's lowers significantly to 0.60 and NoAge-C averages at 0.65. We can also observe that this results are mostly caused by the lower scores obtained in the several target categories; both b_none and non-biased, while also obtaining lower F1-scores than their Group A counterparts, are still significantly higher than the remaining class scores.

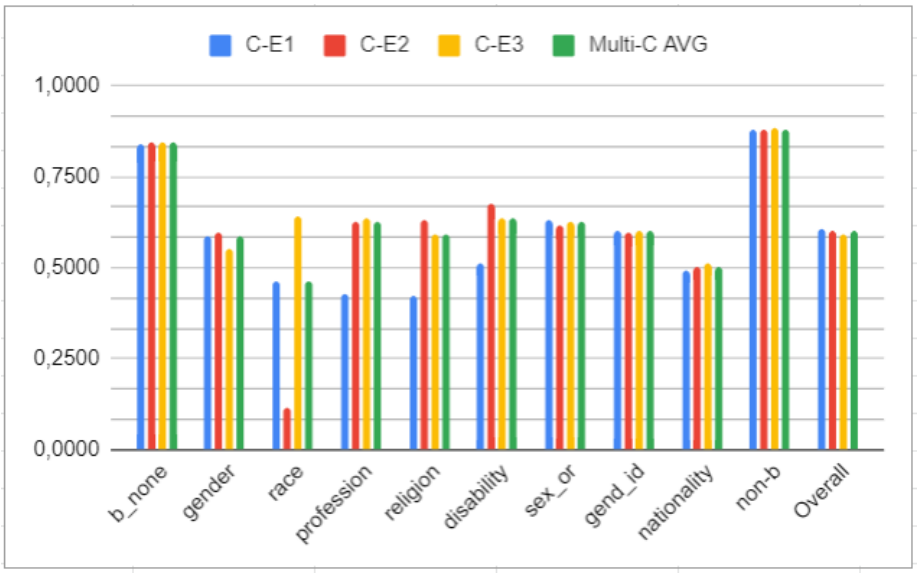

Figure 5.6: Class breakdown of the F1-scores obtained across Multi-C experiments

Figure 5.6 represents the results obtained in the Multi-C experiments. Since the results obtained in the "Age" category are not only null but have also been discussed, we have excluded them from the present analysis. We can observe, in Figure 5.6, some interesting patterns in the results obtained for the several categories. Firstly, there is the anomalous result obtained from the race class, previously mentioned in Section 5.2.2, which was not recurrent.

Secondly, we can also observe that sexual_orientation and gender_identity are the only categories that achieve a F1-score equal to (or greater than) 0.6 across all three experiments. This is rather interesting since these categories make up 1% and 0.61%, respectively, of the total data in Group C, yet achieve better performance than other categories, which leads us to believe that the language found in entries of these types might differ enough from the rest to lead to this result. Curiously, as shown in Annex, sexual_orientation achieves better Precision than Recall, while the opposite is true for Gender Identity. Our hypothesis is that, firstly, slurs and derogatory language related to sexual orientation are frequently used online, and as such might be present in other categories (namely, the unspecified b_none class), thus resulting in a number of sexual_orientation entries being mislabeled as b_none and lowering

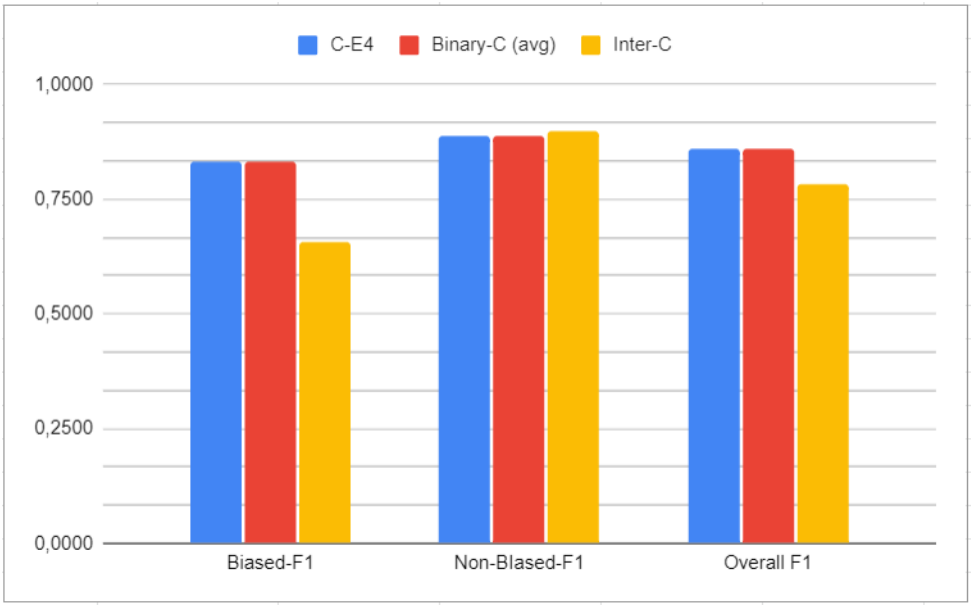

Figure 5.7: F1-scores of experiments C-E4, Inter-C, as well as the average F1-scores of Binary-C

the recall score due to a higher number of false negatives. Gender Identity, however, has only recently become "mainstream"; the likely lack of content regarding this topic in the unspecified biased category, combined with the overall low number of entries labeled as gender_identity and the specificity of this content, might very well result in a higher number of false positives due to overfitting, thus yielding a lower precision score.

The remaining results do not differ as significantly. nationality shows a consistently lower performance than most other classes, but it is also only 0.29% of the total data in Group C, and as such a lower performance is expected.

Lastly, let us examine the results obtained in Binary-C and Inter-C. Figure 5.7 shows the F1-scores obtained for the two classes of the Binary-Classification task (Biased/Non-biased) as well as the overall F1-score. The values in question are those from the Inter-C and C-E4 experiments (models trained during the same number of Epochs and using the same pooling function), as well as the average of all Binary-C experiments.

As shown in Figure 5.7, the biggest difference between the several experiments is the Biased F1-score obtained in Inter-C. It becomes rather clear than while the model trained with Group A data is able to identify Non-Biased entries, with a performance on par with models trained with Group C data, the same cannot be said for biased data. Therefore, we believe that models trained for the Binary-Classification task using general, Twitter-based Bias/Hate Speech Detection datasets do not achieve a satisfactory performance when identifying synthetic/Multi-Target biased content.

In conclusion, it is clear that adding entries labeled for different categories to a general Bias/hate

Speech dataset yields varying results, dependent on the type of language found in each category as well as the overall number of entries for each category; none of these results, however, show a satisfactory performance. We have also discovered that the exclusion of these entries in training results in a similarly unsatisfactory performance. Therefore, while individual category detection is severely lacking, we can conclude that models trained with this type of data succeed in differentiating unspecified biased content from non-biased content.

5.3.3 "Can we obtain a better performance by using all of our resources together?" Or: The Epic of Group D

Lastly, we arrived at our last control group, which is composed by the unification of all our resources. We are, therefore, analysing how well (or how badly) the general, Twitter-based Bias/Hate Speech Detection datasets, Single-Target datasets, and synthetic and/or Multi-Target datasets perform together.

We would like to remind that Group D is the only one to include the CONAN dataset, introduced in Section 2.2 and mentioned in Chapter 4, which is a synthetic, Single-Target dataset for the target category "Religion". This dataset did not fit neatly into the previous control groups, but we decided to nevertheless include it in this Group; "all of our resources", after all, means *all* of our resources.

As can be seen in Table 5.6, we once more find a significant difference in performance between Group-A and Multi-D, partially bridged by NoAge-D. Group A's overall F1-score consistently hits the 0.89 range, while Multi-D's rests in the 0.61 range and NoAge-D falls, on average, in the 0.67 range. Multi-D sees a decrease in performance for both the b_none and non-biased categories, even when compared to Multi-C.

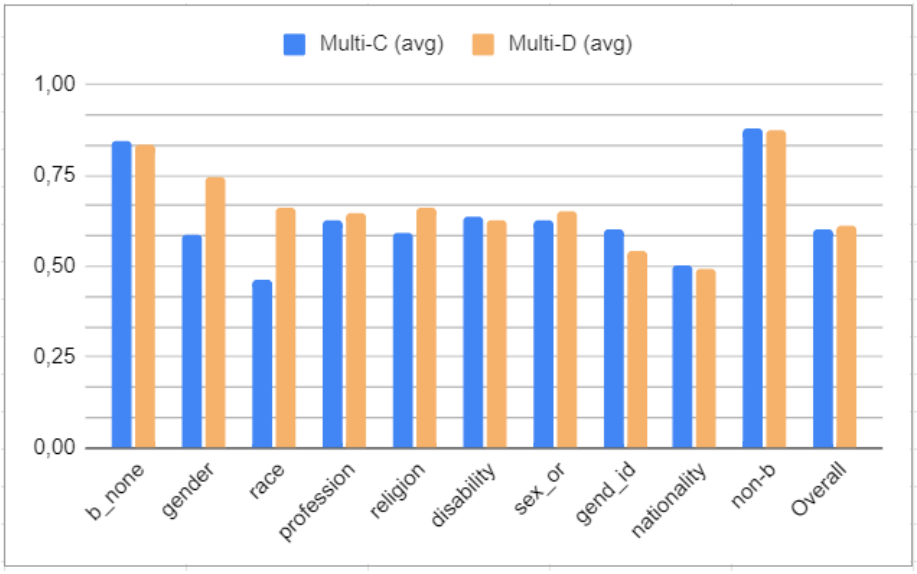

Figure 5.8: Comparison between F1-score averages of Multi-C and Multi-D

Figure 5.8 shows the comparison between F1-scores obtained across all categories for both Multi-C and Multi-D. As previously mentioned, there is a slight decrease in performance for classes b_none and non-biased, which is interesting not due to the severity of the decrease – which, as mentioned, is slight – but due to the fact that it happens at all.

There is, however, a severe decrease in performance worthy of note in gender_identity. We believe this might either be due to the split between train, validation, and test sets – seeing as this class makes up a mere 0.56% of Group D, and, as such, is easily affected by the random data split – or due to some type of overlap of terms with the added gender entries from the Single-Target datasets. The fact that there is no pattern in terms of Precision and Recall, in opposition to what we observed in the previous section, leads us to believe that the first option is the more likely answer.

On the flip side, we also see a noticeable improvement in gender, religion, and race. The latter can, once more, be justified by the lower average value resulting from the anomalous result obtained in Multi-C, rather than any real improvement in the model's behaviour.

The improvement observed in the other two classes, however, we attribute directly to the addition of the Single-Target datasets which deal precisely with the target categories in question. The fact that the improvement in religion is markedly lower than in gender also supports this theory; religion makes up 1.43% of Group C's data compared to 1.56% of Group D, while gender goes from a modest 1.72% in Group C to a respectable 4.26% in Group D. Furthermore, we observed in Section 5.3.1 that the Single-Target entries for "Gender" behaved extremely well when added to the baseline datasets, which we attributed partly to the fact that all datasets in Group B were Twitter-based. CONAN's synthetic origin could be a contributing factor, in addition to the lower overall amount of entries, to the less marked improvement in performance.

Shifting our attention to the Binary-Classification task, Figure 5.9 shows the F1-scores for the biased and non-biased classes, as well as the overall F1-score, obtained in Inter-D and Binary-D, the latter of which is depicted both through the average results of all Binary-D experiments, as well as through the experiment counterpart to Inter-D (trained during the same number of Epochs and using the same pooling function), namely D-E4.

Much like the pattern observed in the previous section, regarding Group C, it becomes clear that the model trained with Group A data, used in the Inter-D experiment, does not perform nearly as well as the model trained with Group D data. Notably, it also performs noticeably worse in the Biased class, which can certainly be attributed to the synthetic and Multi-Target datasets featured in Group D, introducing not only new linguistic forms but also different ways to express and convey bias.

In the end, all our resources together do not perform better than our baseline group. We do observe marked improvement in the classes to which we added new entries when compared to those same classes in Group C, which suggests that the lower performance score might be due to the low number

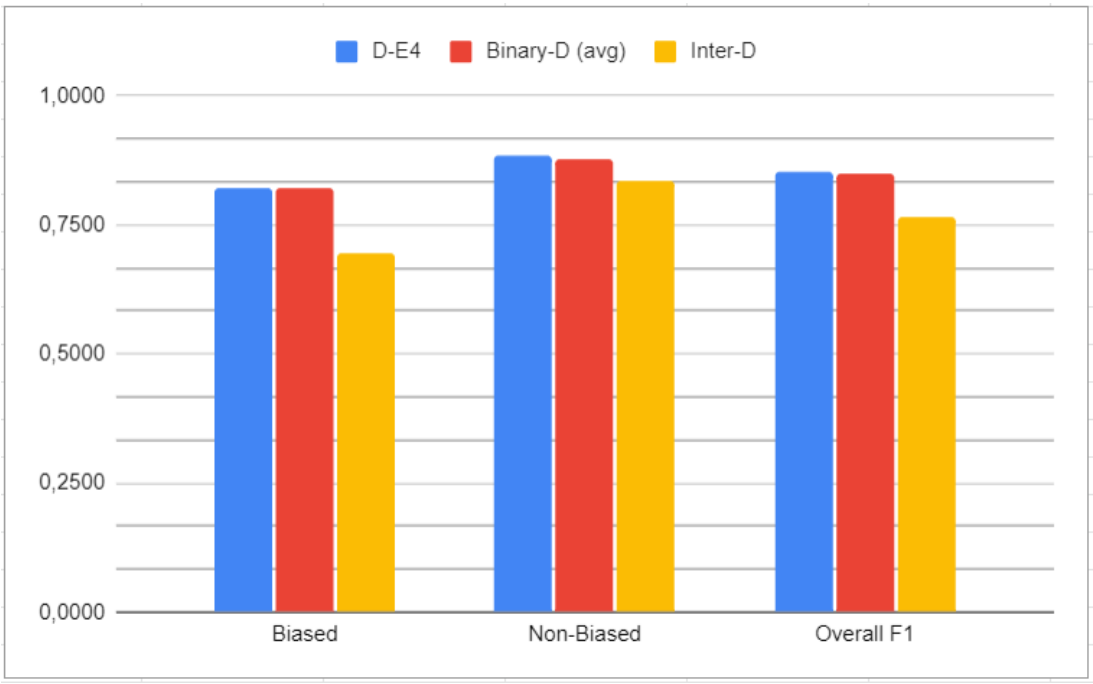

Figure 5.9: F1-scores of experiments D-E4, Inter-D, as well as the average F1-scores of Binary-D

of entries for the several categories rather than the model's inherent difficulty in dealing with the different kinds of biases and categories. Additionally, while the synthetic datasets do not perform as well when the baseline data is Twitter-based, their addition to the training data is markedly necessary if we want a model that can properly identify them, as shown in the comparison between Inter-D and Binary-D.

5.3.4 Conclusion and Next Steps

With these experiments, we have learnt the following:

- Models can learn to identify bias for a certain target category if trained with general/unspecified Bias/Hate Speech Detection datasets and a smaller number of entries labeled for a single category;
- The quantity of entries necessary to obtain a satisfactory performance may depend on whether these entries obey similar linguistic conventions as the general Bias/Hate Speech Detection datasets and/or if the utilized language is often found in the general datasets;
- Models trained solely on Twitter-based datasets do not perform as harmoniously when trained with joint synthetic and Twitter-based datasets;
- However, models trained with purely Twitter-based datasets seem to be worse at recognizing synthetic text, or more nuanced forms of bias, than models trained with joint datasets.

Having derived our conclusions, and, as such, answered our research questions, it is now time for the next part of our work. We must now peruse the obtained results and choose, based on those results, which experiment we want to replicate and utilize in our chosen downstream task – namely, to detect biased content in datasets used Dialogue Models. After some consideration, we chose the D-E10 and D-E4 experiment to proceed on to the next phase of our work.

This choice was motivated by a number of factors. Firstly, because the set of experiments we were most curious to see in practice were those which used all of our resources as training data – in other words, those trained with Group D. When faced with the choice between NoAge-D, Multi-D, and Binary-D, we immediately dismissed Multi-D, since the only significant difference between Multi-D and NoAge-D was the age category, which did not meaningfully contribute to the behaviour of the models in question. Between NoAge-D and Binary-D, we decided to prioritize NoAge-D (and, thus, be fully confronted with how the contradictions between classes and linguistic features may translate to a practical task) and utilize Binary-D only if, after analysing the results obtained by using NoAge-D, we felt the need for more data.

Within the NoAge-D set of experiments, we only had immediate access to the D-E10 model. Since we were facing some temporal constraints, and the results obtained by D-E10 and D-E11 were extremely similar, we decided to proceed with D-E10. Within the Binary-D set of experiments, we picked the model with the best Overall F1-score, namely D-E4.

It is now time, therefore, for the next and final phase of our work.

6

Practical Application

Contents

In this chapter we will introduce the downstream task defined as the end goal for our developed models. We will present, discuss, and analyse the results achieved by our models in this task, as well as compare them with the results obtained in the previous chapter.

OpenSubtitles [6] is a comprehensive collection of subtitles, obtained from a large database of movies and series, spanning several decades, languages, and genres. Subtitles are frequently utilized as training data for Dialogue Models [8], thus making OpenSubtitles the ideal training ground for our work.

We intend to utilize the B-Subtle framework, described in Section 3.4, to obtain and process the subtitles from a previously established selection of movies and series. Afterwards, we will run our chosen model over the collected data and analyse the obtained results, in order to ascertain how well our model behaves in practice.

6.1 Processing OPUS

Since we trained our model on English data and the B-Subtle framework further requires untokenized data, we chose to download the English untokenized version of the corpus, through OPUS [7].[1][2]

After some consideration, and knowing that we were handling a significantly large amount of data, we decided to restrict our research to two categories of movies and shows. Namely, those from the "Animation" genre, released between 2010-2017, and those from the "Comedy" genre, released between 2010-2017. We would like to clarify that movies and shows under the "Animation" genre are not necessarily the same as those in the "Family" genre, since the latter will generally consist of movies aimed at younger audiences, and the former may include any type of content as long as it is animated. This includes, for example, shows such as "The Simpsons", "Family Guy", or "American Dad", which are notably not made for a younger audience.

This selection was motivated by the fact that these two genres frequently host content that is proudly irreverent or satirical, toeing the line of acceptable offense, and thus prone to exhibiting the type of language we mean to target with our work. The temporal selection was motivated by the sociocultural shifts observed in the decade of 2010 to 2020, which was characterized by a growing awareness of how Bias and Hate Speech can manifest, how that can or should impact the way we express ourselves, or the media we consume. Since OPUS only includes titles produced until 2018, and since there is a small collection of titles produced in that year, we decided to restrict our selection from 2010 to 2017.

In order to obtain our selection, we resorted to B-Subtle's Metadata Filters, namely the genre filter, which allowed us to select movies or shows which belonged to the aforementioned genres, and the year filter, which allowed us to select movies or shows released in the the intended year range.

[1] https://opus.nlpl.eu/OpenSubtitles.php
[2] http://www.opensubtitles.org/

Once we were in possession of the relevant files, we converted the files from JSON file type to CSV files. Furthermore, we also combined the individual files into larger files, one for each year. Each entry in the CSV file consists of (examples obtained from an entry from the 2011 Animation file):

- A number-based entry ID characteristic of each year file (NUMBER). For example, *251*
- The ID of the original subtitle file each entry originates from (FILE_ID). For example, *4045346*
- The ID of the of the entry in its original subtitle file (FILE_NUM). For example, *249*
- The textual content of each entry, which was identified as an individual line in the original subtitle file (TEXT). For example, *Don't tell a chicken when to cluck.*

Once we finished this process, we realized that we had too many total entries for the Comedy genre and, subsequently, that the files were too large for our model to evaluate in a timely manner. Therefore, in order to even the training ground between the experiments of both movie genres, we averaged the total number of entries per file of the Animation genre, and then randomly selected that very same number of entries from each file of the Comedy genre. This resulted in a more manageable collection size-wise.

Finally, we had a total of 16 files, 8 for each genre, ready to be tested by our chosen model.

6.2 Initial Classification Results

Each year file was used as testing data, thus allowing us to easily separate our results. We ran the D-E10 model over the entire subtitle data collection and saved the model's predictions. The obtained results are depicted in Tables 6.1 and 6.2. We would like to point out that the "Number of Entries" column in Table 6.2 refers to the aftermath of the reduction mentioned in the previous section, hence the identical numbers.

After a brief perusal of results, we realized that the model tended to classify one word text entries as biased; this was due to the model's initial label probability distribution for each entry, which would be adjusted as it processed the rest of the sentence. Since one-word entries, by definition, did not go through this adjustment process, we found that there was a disproportionate amount of these entries classified by the model as "biased" when compared to other entries. Therefore, we chose to remove one-word entries classified as "biased", and not consider them going forward.

An initial analysis of these results shows that both genres, Animation and Comedy, are characterized by a similar percentage of entries classified as "biased"; more specifically, biased entries make up 1.55% and 1.78% of the Animation and Comedy genres, respectively. However, once we proceeded with the aforementioned removal of one-word entries, that figure changed drastically.

Files	Number of Entries	Non-Biased	Biased			Non-Biased (%)	Biased (%)		
			Total	1 Word	>1 Word		Total	1 Word	>1 Word
2010	325,777	319,921	5,856	424	5,432	98.20%	1.80%	0.13%	1.67%
2011	341,781	336,269	5,512	288	5,224	98.39%	1.61%	0.08%	1.53%
2012	361,614	356,003	5,611	241	5,370	98.45%	1.55%	0.07%	1.49%
2013	363,206	357,770	5,436	281	5,155	98.50%	1.50%	0.08%	1.42%
2014	370,996	365,233	5,763	280	5,483	98.45%	1.55%	0.08%	1.48%
2015	338,732	334,080	4,652	272	4,380	98.63%	1.37%	0.08%	1.29%
2016	329,289	324,316	4,973	212	4,761	98.49%	1.51%	0.06%	1.45%
2017	214,084	210,845	3,239	192	3,047	98.49%	1.51%	0.09%	1.42%
Total	2,645,479	2,604,437	41,042	2,190	38,852	98.45%	1.55%	0.08%	1.47%

Table 6.1: Animation Movies and Shows Subtitles: Statistics and Initial Results with D-E10

	Number of Entries	Non-Biased	Biased			Non-Biased (%)	Biased (%)		
			Total	1 Word	>1 Word		Total	1 Word	>1 Word
2010	340,257	333,978	6,279	5,443	836	98.15%	1.85%	1.60%	0.25%
2011	340,257	334,210	6,047	5,331	716	98.22%	1.78%	1,57%	0.21%
2012	340,257	334,154	6,103	5,374	729	98.21%	1.79%	1.58%	0.21%
2013	340,257	334,285	5,972	5,289	683	98.24%	1.76%	1.55%	0.20%
2014	340,257	334,109	6,148	5,359	789	98.19%	1.81%	1.57%	0.23%
2015	340,257	334,231	6,026	5,333	693	98.23%	1.77%	1.57%	0.20%
2016	340,257	334,240	6,017	5,215	802	98.23%	1.77%	1.53%	0.24%
2017	340,257	334,461	5,796	4,314	1,482	98.30%	1.70%	1.27%	0.44%
Total	2,722,056	2,673,668	48,388	41,658	6,730	98.22%	1.78%	1.53%	0.25%

Table 6.2: Comedy Movies and Shows Subtitles: Statistics and Initial Results with D-E10

As shown in Tables 6.1 and 6.2, entries classified as biased, and which had more than one word, made up for 1.47% of entries in the Animation corpus – a result that is relatively similar to the previously noted 1.55% – and for a mere 0.25% of the Comedy corpus, which a much more significant difference.

From these preliminary results, we can conclude that the Animation corpus contains a higher percentage of biased content when compared to the Comedy genre. Nevertheless, that percentage is still quite low, thus leading us to believe that these corpora are mostly non-biased.

However, we must bear in mind that this is the first time we use our model in a practical task. We must, therefore, evaluate these preliminary results and calculate our model's Accuracy in a practical task, before we draw any further conclusions.

6.3 Result Evaluation

6.3.1 Evaluation Method

In order to evaluate the obtained results, we randomly selected a sample of 75 biased entries from each year file. This brings us to a total of 1200 randomly selected entries; 600 from the Animation corpus, evenly distributed between 3 annotators, and 600 from the Comedy corpus, distributed in the same way. The annotators involved in this work are familiar with Computational Linguistics and NLP in general.

In both cases, entries were divided as follows: each annotator was assigned 50 entries out of the aforementioned 75. These entries purposefully overlapped with the entries assigned to the other annotators, so that every entry would be annotated by 2 annotators. In total, each annotator would deal with a total of 400 entries.

Annotators were given an Annotation Guide, which is included in Appendix B, and asked to review the sentences assigned to them and to classify them in accordance to the definition of Bias adopted in this work and described in Section 2.3. In other words, annotators should label a sentence as biased if:

- The sentence contained a derogatory term or slur which specifically refers to one of our target categories. We will refer to this as a Type 1 biased sentence;
- The sentence expressed a stereotype or caricature referring to one of our target categories. We will refer to this as a Type 2 biased sentence;
- The sentence included otherwise abusive language which specifically targets a group or an individual in our pre-defined target categories. We will refer to this as a Type 3 biased sentence.

Furthermore, in addition to simply classifying sentences as biased or non-biased (which we will henceforth refer to as Label), annotators were also asked to classify the sentence's Type and Category. The former refers to the Types described above; the latter refers to the target categories espoused in our work. Additionally, in the case of non-biased sentences, Type and Category should take values 0 and None, respectively.

In the following Subsections, we will present and analyse statistics and results solely related to Label annotations, since understanding whether or not our model is able to recognize and correctly classify biased content is our biggest concern. In a later section, we will analyse annotator's responses in regards to Type and Category, in order to obtain some insight regarding our Bias Definition when applied in a practical task.

6.3.2 Calculating Inter-Annotator Agreement

We will calculate Inter-Annotator Agreement (IAA) solely in regards to Label. We opted to calculate IAA using Cohen-Kappa Coefficient (k), Pearson Correlation Coefficient *(r, ρ-value)*, and Raw Agreement (*R.A.*). The justification behind calculating R.A. is simple: neither of the chosen coefficients knows how to deal with one or both annotators assigning the same label to all sentences.

Pearson Correlation is calculated as follows:

$$\rho(x, y) = \frac{\sigma(X, Y)}{\sigma_x \sigma_y} \tag{6.1}$$

This means that if one of the variables in question has no variation – or, in other words, is actually a *constant* – then not only is the resulting covariance zero, as also, per the formula, we will be attempting to divide by zero. This results in an undefined value and, thus, does not tell us anything meaningful.

Cohen-Kappa, on the other hand, is defined as:

$$k = \frac{\rho_o - \rho_e}{1 - \rho_e} \tag{6.2}$$

In which ρ_0 is the relative observed agreement between annotators and ρ_e is the hypothetical probability of chance agreement. If one of the annotators assigns the same label to all entries – resulting in constant values for a given variable – then the relative observed agreement and the hypothetical probability of chance agreement are the same, resulting in $k = 0$. Once more, this is not a meaningful result in terms of understanding IAA.

The results are depicted in Tables 6.3 and 6.4. We calculate the aforementioned metrics for each individual year and in total. As we can see, the R.A. for all annotator pairs repeatedly fell in the 0.90 range, which is extremely high and in direct contrast with slightly lower values obtained for k and r – that is, when we obtain values for k and r at all.

As seen in Table 6.3, the k values for the Animation corpus vary between the 0.5 and 0.7 ranges, which indicates moderate to substantial agreement [64]. The Comedy corpus, depicted in Table 6.4, proves to be significantly less straightforward, with one annotator pair, A1 and A3, yielding $k = 0$ due to the previously described issues with constant values in Cohen-Kappa. A1 and A2 reach substantial agreement, with $k = 0.80$, and A1 and A3 reach only a fair agreement, with $k = 0.40$. Once more, this shows a significant contrast against the values for R.A. We hypothesize that this discrepancy may due to the same issue mentioned above; namely, the high prevalence of the same value for one of the variables is interpreted as chance and causes the k to yield lower values.

In conclusion, common metrics show that our annotators reach a fair to substantial level of Inter-Annotator Agreement. However, since this value seems to be unfairly influenced by class imbalance and the issue with constant values for one or both of our chosen variables, we also showcase the Raw Agreement between annotators. Although this metric is not as trustworthy as Cohen-Kappa or Pearson Correlation, and as such is not our main focus, it serves to illustrate that the lower results obtained in these metrics (or not obtained, as the case may be) may be skewed, thus leaving us with a satisfactory level of Inter-Annotator Agreement.

Files	A1 & A3				A1 & A2				A2 & A3			
	k	***R.A.***	***r***	***p-value***	***k***	***R.A.***	***r***	***p-value***	***k***	***R.A.***	***r***	***p-value***
2010	0.8344	0.9600	0.8461	9.93E-08	0.8344	0.9600	0.8461	9.93E-08	1.0000	1.0000	1.0000	0.00E+00
2011	0.7525	0.9200	0.7766	5.00E-06	0.0000	0.9200	-	-	0.7788	0.9600	0.7985	1.71E-06
2012	0.1722	0.8000	0.1746	4.04E-01	0.3590	0.8800	0.4677	1.84E-02	0.4681	0.9200	0.5528	4.16E-03
2013	0.7059	0.9200	0.7385	2.49E-05	0.6479	0.9600	0.6922	1.26E-04	0.4565	0.9200	0.4565	2.18E-02
2014	0.5033	0.8800	0.5104	9.14E-03	0.6479	0.9600	0.6922	1.26E-04	0.7788	0.9600	0.7985	1.71E-06
2015	0.4565	0.9200	0.4565	2.18E-02	0.0000	0.8800	-	-	1.0000	1.0000	1.0000	0.00E+00
2016	0.0000	0.8800	-	-	0.8344	0.9600	0.8461	9.93E-08	0.6269	0.9200	0.6757	2.10E-04
2017	0.6479	0.9600	0.6922	1.26E-04	0.0000	1.0000	-	-	0.0000	1.0000	-	-
Total	0.5583	0.9050	0.5597	6.95E-18	0.5731	0.9400	0.6338	7.36E-24	0.7436	0.9600	0.7577	1.47E-38

Table 6.3: Inter-Annotator Agreement for the Animation Corpus classified by D-E10

Files	A1 & A3				A1 & A2				A2 & A3			
	k	***R.A.***	***r***	***p-value***	***k***	***R.A.***	***r***	***p-value***	***k***	***R.A.***	***r***	***p-value***
2010	0.0000	1.0000	-	-	1.0000	1.0000	1.0000	0.00E+00	0.0000	1.0000	-	-
2011	0.0000	1.0000	-	-	1.0000	1.0000	1.0000	1.59E-181	0.0000	1.0000	-	-
2012	0.0000	0.9600	-	-	0.0000	0.9600	-	-	0.0000	0.9600	-	-
2013	0.0000	1.0000	-	-	0.0000	1.0000	-	-	0.0000	1.0000	-	-
2014	0.0000	1.0000	-	-	0.0000	1.0000	-	-	0.6479	0.9600	0.6922	1,26E-04
2015	0.0000	1.0000	-	-	0.0000	1.0000	-	-	0.0000	0.9600	-	-
2016	0.0000	1.0000	-	-	0.0000	1.0000	-	-	0.0000	1.0000	-	-
2017	0.0000	0.9600	-	-	0.0000	1.0000	-	-	0.0000	1.0000	-	-
Total	0.0000	0.9900	-	-	0.7976	0.9950	0.8144	1.09E-48	0.3952	0.9850	0.4962	7.90E-14

Table 6.4: Inter-Annotator Agreement for the Comedy Corpus classified by D-E10

6.3.3 Accuracy: D-E10, D-E4, and A-E1

After reviewing our annotator's responses, we obtain the results depicted in Tables 6.5 and 6.6.

The tables show the number of entries which were annotated with the same label by the two annotators assigned to those entries (Label Agreement) as well as those in which the annotators disagreed (Label Disagreement). As shown in both Tables, the number of entries annotated with the non-biased label is extremely high. This is startling, since these entries were randomly selected from a collection of entries classified by the model as "biased".

Files	Sample	Label Agreement		Label Disagreement	Accuracy
		Biased	Non-Biased		
2010	75	9	64	2	0.120
2011	75	6	64	5	0.080
2012	75	3	62	10	0.040
2013	75	5	65	5	0.067
2014	75	5	65	5	0.067
2015	75	3	67	5	0.040
2016	75	5	64	6	0.067
2017	75	1	73	1	0.013
Total	600	37	524	39	0.062

Table 6.5: Accuracy for D-E10 on the Animation Corpus

Files	Sample	Label Agreement		Label Disagreement	Accuracy
		Biased	Non-Biased		
2010	75	1	74	0	0.013
2011	75	1	74	0	0.013
2012	75	0	72	3	0.000
2013	75	0	75	0	0.000
2014	75	1	73	1	0.013
2015	75	0	74	1	0.000
2016	75	0	75	0	0.000
2017	75	0	74	1	0.000
Total	600	3	591	6	0.005

Table 6.6: Accuracy for D-E10 on the Comedy Corpus

Accuracy for both models is extremely low, notably so in the case of the Comedy corpus, in which only 3 entries, out of 600, were labeled by both annotators as "biased". The F1-score obtained in Model Testing for D-E10 was in the 0.6 range (found in Table 5.12), which, although somewhat low, is a result that does not match the model's Accuracy in this task. This discrepancy reveals that our model, quite simply, does not work as intended when used in a practical task.

This realization made us question the performance of our other developed models. As mentioned in Section 5.3.4, we intended to use model D-E4 if we found that we required more data to properly understand how these models behaved in a practical task. Having verified this condition, we repeated some

of our experiments using the the D-E4 model. D-E4, trained during 4 epochs and with the avg pooling function, yielded the best result out of the Binary-D experiments, achieving an F1-score of 0.8515 (as depicted in Table 6.8). Additionally, we decided to also use the model with the best overall performance, namely A-E1 (F1-score of 0.8974, found in Table 5.4), and observe how it behaves in a practical task.

Due to temporal restraints, we ran the A-E1 model over the Animation corpus and the D-E4 model over the Comedy corpus, rather than run both models over both corpora. We believe that repeating the experiment would yield similar results, which would be redundant and not paramount for our work.

Files	Number of Entries	Non-Biased	Biased			Non-Biased (%)	Biased (%)		
			Total	1 Word	>1 Word		Total	1 Word	>1 Word
2010	325,777	319,534	6,243	206	6,037	98.08%	1.92%	1.85%	0.06%
2011	341,781	335,962	5,819	115	5,704	98.30%	1.70%	1.67%	0.03%
2012	361,614	355,657	5,957	77	5,880	98.35%	1.65%	1.63%	0.02%
2013	363,206	357,994	5,212	101	5,111	98.57%	1.43%	1.41%	0.03%
2014	370,996	365,456	5,540	151	5,389	98.51%	1.49%	1.45%	0.04%
2015	338,732	333,986	4,746	101	4,645	98.60%	1.40%	1.37%	0.03%
2016	329,289	324,087	5,202	93	5,109	98.42%	1.58%	1.55%	0.03%
2017	214,084	210,765	3,319	38	3,281	98.45%	1.55%	1.53%	0.02%
Total	2,645,479	2,603,441	42,038	882	41,156	98.41%	1.59%	1.56%	0.03%

Table 6.7: Animation Corpus tested with A-E1: Statistics and Initial Results

Files	Number of Entries	Non-Biased	Biased			Non-Biased (%)	Biased (%)		
			Total	1 Word	>1 Word		Total	1 Word	>1 Word
2010	340,257	333,978	6,279	5,250	1,029	98.15%	1.85%	0.30%	1.54%
2011	340,257	334,210	6,047	5,117	930	98.22%	1.78%	0.27%	1.50%
2012	340,257	334,154	6,103	5,283	820	98.21%	1.79%	0.24%	1.55%
2013	340,257	334,285	5,972	5,149	823	98.24%	1.76%	0.24%	1.51%
2014	340,257	334,109	6.148	5,225	923	98.19%	1.81%	0.27%	1.54%
2015	340,257	334,231	6,026	5,136	890	98.23%	1.77%	0.26%	1.51%
2016	340,257	334,240	6,017	5,115	902	98.23%	1.77%	0.27%	1.50%
2017	340,257	334,461	5,796	4,038	1,758	98.30%	1.70%	0.52%	1.19%
Total	2,722,056	2,673,668	48,388	40,313	8,075	98.22%	1.78%	0.30%	1.48%

Table 6.8: Comedy Corpus tested with D-E4: Statistics and Initial Results

We followed the same procedure described in previous sections, for processing, testing, and evaluation. Tables 6.7 and 6.8 show the results from running the A-E1 and D-E4 models on the Animation and Comedy corpora, respectively. These results are similar to their counterparts, shown in Tables 6.1 and 6.2, in the sense that, once more, the Animation corpus has a higher count of entries marked as biased and with more than one word, while the Comedy corpus not only has a lower count overall, but most of those entries classified as "biased" are 1-word entries which will be removed. Both A-E1 and D-E4 classify more sentences as biased overall, when in comparison with their D-E10 counterparts.

Regarding the annotation procedure, the biased entries were once again randomly sampled, yielding a total of 75 sentences per year file. These were evenly distributed across annotators, with 3 annotators per corpus. The IAA for these experiments can be found in Tables 6.9 and 6.10, while the Accuracy resulting from the annotator's responses can be found in Tables 6.11 and 6.12.

Files	A1 & A3				A1 & A2				A2 & A3			
	k	***R.A***	***r***	***p-value***	***k***	***R.A***	***r***	***p-value***	***k***	***R.A***	***r***	***p-value***
2010	0.8344	0.9600	0.8461	9.93E-08	1.0000	1.0000	1.0000	0.00E+00	0.7826	0.9200	0.8018	1.45E-06
2011	0.4681	0.9200	0.5528	4.16E-03	0.7059	0.9200	0.7385	2.49E-05	0.0000	0.9600	-	-
2012	0.5161	0.8800	0.5898	1.92E-03	0.7788	0.9600	0.7985	1.71E-06	0.8837	0.9600	0.8898	2.68E-09
2013	1.0000	1.0000	0.0000	0.00E+00	0.6479	0.9600	0.6922	1.26E-04	1.0000	1.0000	1.0000	4.59E-178
2014	1.0000	1.0000	1.0000	0.00E+00	0.7788	0.9600	0.7985	1.71E-06	1.0000	1.0000	1.0000	0.00E+00
2015	0.8344	0.9600	0.8461	9.93E-08	0.6479	0.9600	0.6922	1.26E-04	-0.0417	0.9200	-0.0417	8.43E-01
2016	0.6032	0.8800	0.6571	3.59E-04	1.0000	1.0000	1.0000	0.00E+00	1.0000	1.0000	1.0000	1.59E-181
2017	0.6212	0.9200	0,6212	9.19E-04	0.7024	0.9200	0.7024	9.07E-05	1.0000	1.0000	1.0000	0.00E+00
Total	0.7162	0.9400	0.7194	3.60E-33	0.7959	0.9600	0.8001	8.03E-46	0.8674	0.9700	0.8683	3.47E-62

Table 6.9: Inter-Annotator Agreement for the Animation Corpus classified by A-E1

Files	A1 & A3				A1 & A2				A2 & A3			
	k	***R.A***	***r***	***p-value***	***k***	***R.A***	***r***	***p-value***	***k***	***R.A***	***r***	***p-value***
2010	0.0000	1.0000	-	-	0.0000	1.0000	-	-	0.6479	0.9600	0.6922	1.26E-04
2011	0.0000	1.0000	-	-	0.0000	0.9600	-	-	0.0000	1.0000	-	-
2012	0.0000	1.0000	-	-	0.0000	1.0000	-	-	0.0000	1.0000	-	-
2013	0.0000	1.0000	-	-	0.0000	1.0000	-	-	1.0000	1.0000	1.0000	1.59E-181
2014	1.0000	1.0000	1.0000	0.00000	0.0000	1.0000	-	-	0.0000	1.0000	-	-
2015	1.0000	1.0000	1.0000	0.00000	-0.0417	1.0000	-0.0417	8.43E-01	0.0000	0.9600	-	-
2016	0.0000	1.0000	-	-	1.0000	1.0000	1.0000	1.59E-181	0.0000	0.9200	-	-
2017	0.0000	1.0000	-	-	0.0000	1.0000	-	-	0.0000	1.0000	-	-
Total	0.6622	1.0000	0.7035	3.42E-31	0.3927	0.9950	0.4010	3.99E-09	0.4904	0.9800	0.50718	1.04E-14

Table 6.10: Inter-Annotator Agreement for the Comedy Corpus classified by D-E4

The IAA obtained from the A-E1 run on the Animation corpus is high for all annotator pairs, from substantial to "almost perfect" in the A2 and A3 pair, which reached $k = 0.87$. The same cannot be said for D-E4 on the Comedy corpus, which is more inconsistent. The first annotator pair yields substantial agreement ($k = 0.66$), with the other two reaching fair ($k = 0.40$) to moderate agreement ($k = 0.49$).

Raw Agreement, additionally, falls over the 0.90 range for all annotator pairs, including a perfect agreement percentage for the A1 and A3 pair of the Comedy corpus.

Files	Sample	Label Agreement		Label Disagreement	Accuracy
		Biased	Non-Biased		
2010	75	11	61	3	0.147
2011	75	4	66	5	0.053
2012	75	9	61	5	0.120
2013	75	5	69	1	0.067
2014	75	5	69	1	0.067
2015	75	4	67	4	0.053
2016	75	10	62	3	0.133
2017	75	11	60	4	0.147
Total	600	59	515	26	0.098

Table 6.11: Accuracy for A-E1 on the Animation Corpus

Files	Sample	Label Agreement		Label Disagreement	Accuracy
		Biased	Non-Biased		
2010	75	1	73	1	0.013
2011	75	0	74	1	0.000
2012	75	0	75	0	0.000
2013	75	1	74	0	0.013
2014	75	1	74	0	0.013
2015	75	2	72	1	0.027
2016	75	1	72	2	0.013
2017	75	0	75	0	0.000
Total	600	6	589	5	0.010

Table 6.12: Accuracy for D-E4 on the Comedy Corpus

Finally, Tables 6.11 and 6.12 breakdown the updated annotation results. Once more, we observe a trend: the Animation corpus obtains a much higher Accuracy score than the Comedy corpus, but, in both cases, that score is extremely low. Therefore, we can conclude that it is not only D-E10 which performs poorly in this practical task; rather, all of our models do.

6.4 Interlude: Type and Category

As stated in Section 6.1.3, we also analysed how annotators classified entries in regards to Type and Category. This analysis does not inform our analysis of model performance as much as it does our proposed definition of Bias. More specifically, understanding how annotators interpreted our definition –

and analysing any confusion or problem that may arise therein – helps us to, in turn, understand how to approach Bias in future works.

We began by analysing how annotators labeled for Type and Category for those cases in which both annotators labeled a sentence as Biased. In this situation, there are four possible scenarios:

- Scenario 1: Both annotators agree on Type and Category labels;
- Scenario 2: Annotators agree on Category but not on Type;
- Scenario 3: Annotators agree on Type but not on Category;
- Scenario 4: Annotators disagree onType and Categories

Table 6.13 shows a breakdown of these four scenarios across the four experiments. As we can see, the D-E10 iteration on the Comedy corpus is the only one where Scenario 1 (namely, annotator agreement over both Type and Category) is not the most prevalent scenario. This is clearly the case in the other experiments, with Scenario 1 making up 75.68%, 88.14%, and 66.67% of biased-agreement cases for D-E10 and AE-1 on the Animation corpus and D-E4 on the Comedy corpus, respectively.

	Total		**Scenario 1**		**Scenario 2**		**Scenario 3**		**Scenario 4**	
	Count	**%**	**Count**	**%**	**Count**	**%**	**Count**	**%**	**Count**	**%**
Animation (D-E10)	37	100.00	28	75.68	5	13.51	1	2.70	3	8.11
Comedy (D-E10)	3	100.00	1	33.33	2	66.67	0	0.00	0	0.00
Animation (A-E1)	59	100.00	52	88.14	5	8.47	2	3.39	0	0.0
Comedy (D-E4)	6	100.00	4	66.67	1	16.67	1	16.67	0	0.00

Table 6.13: Type and Category: Agreement Scenarios

Further investigation of Scenario 2 shows that disagreement on Type is slightly higher between Types 2 and 3 (7 occurrences) than between Types 1 and 3 (5 occurrences), with Type 1 and 2 disagreement only being observed in one occasion. As for Scenarios 3 and 4, disagreement on Category is observed either in cases in which annotators felt like there was more than one Target Category found in the entry or, more interestingly, in the Race/Religion/Nationality axis – which is a source of confusion we first mentioned in Chapter 2, after proposing our Bias definition.

Unsurprisingly, the most frequent Target Category found in the annotated entries was Gender, closely followed by Race. Additionally, Type 1 sentences were by far the most frequent. This pattern can be found even in sentences which fell under Label Disagreement, that is, those which were labeled as Biased by only one of the annotators. However, in these sentences we can also find instances of other Target Categories, such as Age, Disability, and Sexual Orientation. A more detailed breakdown of these results is included in Annex.

There are some relevant conclusions to be drawn here. There is no escaping a degree of ambiguity in any Bias definition; after all, not only do these definitions often rely on real-world knowledge (presuming,

therefore, that those interacting with the definition possess that knowledge in the first place) as there is also the matter of personal opinion. If we define Biased language as something that can be “offensive” or “harmful”, we must also contend with the fact that not everyone will apply those terms equally.

Therefore, upon presenting our definition, we were aware of a degree of inescapable ambiguity which we now can find reflected in these results. Biased sentences of Type 1 should have been those that caused the least confusion, since these only deal with the concrete presence or absence of derogatory terms. However, the fact that Type 1 sentences were extremely common in entries only labeled as biased by one annotator leads us to believe that our annotators did not possess a similar and previously established knowledge of certain derogatory terms. While we did provide a number of examples in our Annotation Guide, we believe that a more thorough compendium of terms would greatly benefit in reducing this disagreement.

Additionally, the fact that Target Categories such as Age and Disability were more common in sentences labeled as Biased by one annotator, rather than those labeled Biased by both annotators, can indicate that these are Target Categories towards which biased behaviour is not easily recognized by our annotators. We believe that this could be due to the fact that these are Target Categories that either do not face as overt Bias in real-world contexts, or simply face Bias and are not as frequently discussed. The last option further supports our stance that lesser known forms of Bias are in urgent need of study in the Hate Speech and Bias Detection field.

In conclusion, while our annotators performed well when following our Annotation guide, often agreeing in both Type and Category for those sentences which both annotators labeled as Biased, an avoidable level of ambiguity remains. Improvements to the Annotation Guide, such as the inclusion of a more comprehensive list of possible derogatory terms and stereotypes, would reduce some of this ambiguity. Other influencing factors, however, such as heightened awareness regarding less visible forms of Bias, would require a more complex solution.

6.5 Model Performance: Discussion and Conclusions

We have compiled the most relevant information for our results discussion in Table 6.14. The first column shows the average amount of biased entries with more than 1 word per file, for each of the experiments; the second to fourth columns show an average of the total IAA scores obtained by the three distinct annotator pairs of each experiment; lastly, the fifth column simply shows the Accuracy score obtained for the total sum of files of each experiment. Additionally, in order to calculate the avg r for the D-E10 Comedy experiment, we assigned $r = 0$ to the annotator pair that obtained an undefined result.

There are some insights which we have already mentioned; namely, the fact that the A-E1 model, tested on the Animation corpus, yields the best results on Accuracy, Cohen-Kappa, and Pearson Corre-

lation. It is also the model with the highest count of biased entries. This combination of facts leads us to believe that this is the model that will overall accurately classify the largest amount of biased content.

	Biased (>1 word)	***k***	***R.A***	***r***	**Accuracy**
Animation (D-E10)	5,189.5	0.6250	0.9350	0.6504	0.0617
Comedy (D-E10)	759	0.3976	0.9900	0.4369	0.0050
Animation (A-E1)	5,250	0.7932	0.9567	0.7959	0.0983
Comedy (D-E4)	912.5	0.5151	0.9917	0.5372	0.0100

Table 6.14: Average of results obtained in previous experiments

This is not necessarily a surprise; after all, we defined Group A as our testing baseline because we expected it would perform better than the rest, and A-E1 in particular was the model which achieved the best scores in testing. However, Group A was also never the main focus of our work – hence why it served only as a baseline. It is, however, rather interesting that the model purely trained on Twitter-based data is also the one that performs the best when classifying subtitles. We believe that the way the original JSON files are structured, which leads to a lot of entries featuring incomplete sentences with little context, might have influenced this result.

More interesting is the difference in performance between D-E10 and D-E4 on the Comedy corpus. D-E4 doubles the Accuracy score of D-E10, but these models were trained with the same training and validation data. The only significant difference between them is that D-E10 was trained in the task of Multi-Target Classification and D-E4 was trained in the Binary Classification task. This allows us to conclude that the model's performance is definitely harmed when it attempts to learn the different Target Categories, which have a lot less available entries from which the model can learn from in the first place. Once more, this merely enforces our belief that there is urgent need to create more diverse and inclusive resources, rather than simply directing our attention towards one or two Target Categories which have already been more thoroughly invested in.

Additionally, we calculated the number of sentences which were labeled as biased by both experiment pairs (that is to say, by the pair of experiments conducted on each corpus). We found that there was a significantly higher overlap between the experiments conducted over the Comedy corpus in comparison to those in in the Animation corpus. This translates to 40.86% and 34.06% of all entries classified as biased by D-E10 and D-E4 on the Comedy corpus, respectively, against a mere 12.87% and 12.18% of D-E10 and A-E1, respectively. A proper, sentence-by-sentence analysis of this overlap could yield illuminating results – we will have to, nevertheless, leave that to future work.

There is still more insight to be garnered from these experiments, more in regards to the content of the subtitles themselves. For example, the higher rate of Raw Agreement for both experiments ran over the Comedy corpus is a direct contrast to the Cohen-Kappa and Pearson Correlation Coefficients, but is also a relatively simple phenomenon. Since the Comedy corpus had a higher amount of non-biased

sentences, or sentences in which the bias was less ambiguous, they reached an easier understanding than annotators of the Animation corpus. This supports the hypothesis that subtitles belonging to the Animation genre contain a higher amount of biased – or ambiguously biased – content than those of the Comedy genre.

Lastly, after observing and discussing the obtained results, we may now refer back to the research question motivating this work: "How can pre-existing resources, namely publicly available datasets, be used to train classifiers in the task of Bias Classification – if they can be used to this end at all?".

We can now state that the answer to this question is: "They cannot – or, at least, not in this way."

Evidently, our models failed profusely, even our baseline, which featured a reasonably balanced split between classes, was composed solely of Twitter-based data and thus unlikely to fall prey to issues resulting from being trained with different linguistic conventions, and composed by datasets which generally followed similar conventions and definitions. These were the problems we expected and prepared to tackle when we devised our dataset groups. Evidently, "extremely poor performance in the downstream task" was not one of those problems.

There are a number of concessions that can be made to partially justify this result. After all, we did not set out to build a highly specialized model, and the pre-trained model we did use, namely DistillBERT, is not as good as models such as BERT or RoBERTa. Either one of these changes could, and quite possibly would, have resulted in better performance of the developed models, as well as higher Accuracy.

There are other variables, however, that we can and should question. For example, the usefulness of the datasets we used in this work when used to train models in the sort of task we aimed for – or, even, in any downstream task. After all, we achieved very fair results in terms of precision, recall, and F1-score when we tested our models initially, did we not?

The difference between those high scores and the extremely low Accuracy revealed in this Chapter is, perhaps, the most significant conclusion that we can derive from this work. A notable majority of the datasets we collected, and even of those we found in later research, did not use their datasets in any sort of downstream task. After confronting the results of our work, we truly believe it is paramount for researchers to not only be clear in the downstream tasks they intend to tackle, but also, and most importantly, to take the extra step and properly test their work in the context of that very same task. This would allow researchers to obtain better understanding of their work and, consequently, bring significant advances to any field of study.

7

Conclusion

Contents

7.1 Main Conclusion

Bias in NLP is a recent field of study, with plenty of works being published in recent years. We are discovering that there are many ways in which human biases can, and do, infiltrate our programs and algorithms. One of these ways is through biased training data, which teaches models how to replicate those very same biases.

In our work, we sought to use publicly available resources to train a classifier in the task of Bias Detection and Classification, with the end goal of testing our classifiers in datasets used as training data for Dialogue Models. The aim of our work was to discover if (or how) pre-existing resources could be used together to train a classifier in this task, later classifying Dialogue Model datasets in order to have a concrete downstream task in which to test our model's performance.

In Chapter 1, we shared the motivation behind our work, by introducing some two study cases of Dialogue Systems showcasing biased behaviour. We defined the main problem we wanted to tackle and outlined objectives that would allow us to reach this goal.

In Chapter 2, we defined the concepts of "Bias" and "Hate Speech", emphasizing their inherent dependency on real-life sociocultural dynamics and introducing the concept of *task-specificity*. With this in mind, we presented the definition of Bias developed and adopted in this work. We also presented an ethical statement regarding certain limitations of our work.

Chapter 3 contains an overview of work done in both Bias and Hate Speech Detection, highlighting works with a similarity to ours as well as works which have influenced both fields. We describe datasets and testing approaches developed in the scope of these fields as well as well-known NLP models and algorithms commonly adopted in tasks similar to ours. We also present some critiques of this field of study, which became readily apparent to us after our research. Finally, we introduce the B-Subtle framework, developed for processing of subtitle-based data.

In Chapter 4 we begin by setting the stage for the development of our work. Then, we describe the process of retrieving and processing the data from all of our datasets, as well as the various changes we had to make, such as label mapping. This chapter also touches on the problem of non-persistent data and the consequences of building datasets with data which might become unavailable.

In Chapter 5, we describe our Experimental Setup and analyse the results obtained by our models. We find that while models can learn to identify Bias for a certain target category when trained when unspecified Bias/Hate Speech Detection datasets and a smaller dataset for that very same target category (Single-Target Classification), they do not perform well if one follows this system with a lot of target categories and smaller datasets of varying sizes. However, models trained in this way still appear to be better at identifying Bias in synthetic text, or in more nuanced forms, than datasets trained only on generalized datasets and Twitter-based data, which implies that the model does learn additional information that allows it to perform better in select contexts. The fact that the datasets trained solely on

Twitter-based data did not perform as well in classifying datasets featuring synthetic text implies that linguistic conventions across datasets matter, and that linguistic compatibility may influence how much data is necessary for a model to learn to identify Bias for a given Target Category.

Chapter 6 discusses the performance of some of our models in our chosen downstream task. We find that although the developed models perform well enough to give us some insight regarding which of the tested genres contains more biased content, their performance is overwhelmingly poor. This further solidifies our belief that testing our work in the context of the task that work is developed for is the only way to properly understand what we have created and developed.

This work was fundamentally experimental, and, as such, often sprawling over different types of work done in NLP. In the beginning of our work, we delineated the following objectives:

- Find and collect publicly available datasets aimed at Bias Classification, to serve as training data for our own classifiers;
- Train and analyse the performance of several classifiers, trained with different parameters and training data combinations;
- Run a select few of our developed classifiers over datasets used to train Dialogue Models and then analyse their performance.

Having succeeded at our objectives, we nevertheless conclude that the answer to our research question, namely *"How can pre-existing resources, namely publicly available datasets, be used to train models in the task of Bias Classification - if they can be used to this end at all?"*, is that they cannot be properly utilized for this task; or, at least, not in the way we did.

7.2 Future Work

Due to its experimental nature, there were plenty of decisions made throughout this work that leave us with many possible avenues for further exploration. Such as:

- **Testing our models on the MAIA Customer Service Dataset**. Our work was developed within the scope of the MAIA project, which deals with Customer Service through Dialogue Models. This is, therefore, a type of downstream task which directly interacts with users in real time, sharing the same characteristics which led us to focus on subtitles in the first place. The studies conducted and models developed in this work are suitable to utilize in the scope of the MAIA project;
- **Dataset balancing**. In this work, we purposefully chose not to balance our datasets. This was due to a variety of reasons, further detailed in Section 6.1.2, but most notably because we felt that the best way to balance our datasets, namely Data Augmentation, would not result in faithful depictions

of Bias. However, the obtained results show that the imbalance in our datasets was extremely prejudicial to our final results. Therefore, we believe that developing methods for augmenting existing datasets, in a way that respects the complexities of Bias as a phenomenon, is a very important step to developing better methods of Bias Detection;

- **Development of new model architectures**. In this work, we purposefully chose not to develop our own model architecture, or spend a significant portion of our attention on fine-tuning. The development of an architecture specifically aimed at Bias Detection could optimize the use of pre-existing resources in a completely different way that what we were able to achieve;

- **Different dataset and model combinations**. Less derivative, but still worthy of exploration, would be to train and test with either a different pre-trained Transformer model (such as BERT or RoBERTa) or different combinations of datasets – or both;

- **Multi-labeling**. One of the issues we were confronted with, particularly after reviewing annotator's responses, was the fact that instances of bias regarding more than one Target Category can coexist in text, either separately or through an intersectional lens. Allowing for multi-labeling classification would solve this issue as well as introduce a higher level of intersectionality. As mentioned in chapter 2, this was one of the limitations faced in the current work;

- **Allowing for more than sentence-level classification**. The text we analysed in this work was at Tweet-level or sentence-level. We also chose to remove the contextual aspect of these sentences from our work as much as possible, limiting ourselves to analysing solely biases which were evident at individual sentence-level. This leaves open the possibility of training a model to classify textual instances beyond the sentence-level, thus tackling the difficulty in addressing context which models in this field have been reported to show;

- **Classify Types, not Categories**. An interesting follow-up to our work would be to utilize the same dataset collection while using it to teach models how to detect and classify different types of biases, without differentiating between the categories or social groups targeted by those biases. This could better take advantage of the datasets in question, minimizing the disparity between very different forms of biases (such as, for example, seemingly benevolent stereotypes and openly derogatory comments).

Bibliography

[1] S. Carty, "Many cars tone deaf to women's voices," *AOL Autos*, 2011.

[2] J. A. Rodger and P. C. Pendharkar, "A field study of the impact of gender and user's technical experience on the performance of voice-activated medical tracking application," *International Journal of Human-Computer Studies*, vol. 60, no. 5-6, pp. 529–544, 2004.

[3] T. Bolukbasi, K.-W. Chang, J. Y. Zou, V. Saligrama, and A. T. Kalai, "Man is to computer programmer as woman is to homemaker? debiasing word embeddings," *Advances in neural information processing systems*, vol. 29, pp. 4349–4357, 2016.

[4] A. W. Flores, K. Bechtel, and C. T. Lowenkamp, "False positives, false negatives, and false analyses: A rejoinder to machine bias: There's software used across the country to predict future criminals. and it's biased against blacks," *Fed. Probation*, vol. 80, p. 38, 2016.

[5] M. J. Wolf, K. W. Miller, and F. S. Grodzinsky, "Why we should have seen that coming: comments on microsoft's tay "experiment," and wider implications," *The ORBIT Journal*, vol. 1, no. 2, pp. 1–12, 2017.

[6] P. Lison and J. Tiedemann, "Opensubtitles2016: Extracting large parallel corpora from movie and tv subtitles," 2016.

[7] J. Tiedemann, "Parallel data, tools and interfaces in opus." in *Lrec*, vol. 2012. Citeseer, 2012, pp. 2214–2218.

[8] D. Ameixa, L. Coheur, P. Fialho, and P. Quaresma, "Luke, i am your father: dealing with out-of-domain requests by using movies subtitles," in *International Conference on Intelligent Virtual Agents*. Springer, 2014, pp. 13–21.

[9] M. Ventura, J. Veiga, L. Coheur, and S. Gama, "The b-subtle framework: tailoring subtitles to your needs," *Language Resources and Evaluation*, vol. 54, no. 4, pp. 1143–1159, 2020.

[10] S. Barikeri, A. Lauscher, I. Vulić, and G. Glavaš, "RedditBias: A real-world resource for bias evaluation and debiasing of conversational language models," 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). Online: Association for Computational Linguistics, Aug. 2021, pp. 1941–1955. [Online]. Available: https://aclanthology.org/2021.acl-long.151

[11] B. Vidgen, T. Thrush, Z. Waseem, and D. Kiela, "Learning from the worst: Dynamically generated datasets to improve online hate detection," 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)*. Online: Association for Computational Linguistics, Aug. 2021, pp. 1667–1682. [Online]. Available: https://aclanthology.org/2021.acl-long.132

[12] A. Founta, C. Djouvas, D. Chatzakou, I. Leontiadis, J. Blackburn, G. Stringhini, A. Vakali, M. Sirivianos, and N. Kourtellis, "Large scale crowdsourcing and characterization of twitter abusive behavior," in *Proceedings of the International AAAI Conference on Web and Social Media*, vol. 12, no. 1, 2018.

[13] A. Field, S. L. Blodgett, Z. Waseem, and Y. Tsvetkov, "A survey of race, racism, and anti-racism in NLP," 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)*. Online: Association for Computational Linguistics, Aug. 2021, pp. 1905–1925. [Online]. Available: https://aclanthology.org/2021.acl-long.149

[14] K. Crenshaw, *Demarginalizing the intersection of race and sex: A Black feminist critique of antidiscrimination doctrine, feminist theory, and antiracist politics [1989]*. Routledge, 2018.

[15] C. Basta, M. R. Costa-jussà, and N. Casas, "Evaluating the underlying gender bias in contextualized word embeddings," in *Proceedings of the First Workshop on Gender Bias in Natural Language Processing*, 2019, pp. 33–39.

[16] M. Kaneko and D. Bollegala, "Gender-preserving debiasing for pre-trained word embeddings," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019, pp. 1641–1650.

[17] W. Guo and A. Caliskan, *Detecting Emergent Intersectional Biases: Contextualized Word Embeddings Contain a Distribution of Human-like Biases*. New York, NY, USA: Association for Computing Machinery, 2021, p. 122–133. [Online]. Available: https://doi.org/10.1145/3461702.3462536

[18] M. Jiang and C. Fellbaum, "Interdependencies of gender and race in contextualized word embeddings," in *Proceedings of the Second Workshop on Gender Bias in Natural Language*

Processing. Barcelona, Spain (Online): Association for Computational Linguistics, Dec. 2020, pp. 17–25. [Online]. Available: https://aclanthology.org/2020.gebnlp-1.2

[19] Y. C. Tan and L. E. Celis, "Assessing social and intersectional biases in contextualized word representations," *arXiv preprint arXiv:1911.01485*, 2019.

[20] S. Sharifirad, A. Jacovi, I. B. I. Univesity, and S. Matwin, "Learning and understanding different categories of sexism using convolutional neural network's filters," in *Proceedings of the 2019 Workshop on Widening NLP*, 2019, pp. 21–23.

[21] M. Nadeem, A. Bethke, and S. Reddy, "StereoSet: Measuring stereotypical bias in pretrained language models," 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)*. Online: Association for Computational Linguistics, Aug. 2021, pp. 5356–5371. [Online]. Available: https://aclanthology.org/2021.acl-long.416

[22] P. Parikh, H. Abburi, P. Badjatiya, R. Krishnan, N. Chhaya, M. Gupta, and V. Varma, "Multi-label categorization of accounts of sexism using a neural framework," 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)*, 2019, pp. 1642–1652.

[23] H. Liu, W. Wang, Y. Wang, H. Liu, Z. Liu, and J. Tang, "Mitigating gender bias for neural dialogue generation with adversarial learning," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online: Association for Computational Linguistics, Nov. 2020, pp. 893–903. [Online]. Available: https://aclanthology.org/2020.emnlp-main.64

[24] B. H. Zhang, B. Lemoine, and M. Mitchell, "Mitigating unwanted biases with adversarial learning," in *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, 2018, pp. 335–340.

[25] J. Zhao, T. Wang, M. Yatskar, V. Ordonez, and K.-W. Chang, "Gender bias in coreference resolution: Evaluation and debiasing methods," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, 2018, pp. 15–20.

[26] S. Kiritchenko and S. Mohammad, "Examining gender and race bias in two hundred sentiment analysis systems," in *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*. New Orleans, Louisiana: Association for Computational Linguistics, Jun. 2018, pp. 43–53. [Online]. Available: https://aclanthology.org/S18-2005

[27] A. Garimella, C. Banea, D. Hovy, and R. Mihalcea, “Women’s syntactic resilience and men’s grammatical luck: Gender-bias in part-of-speech tagging and dependency parsing,” in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019, pp. 3493–3498.

[28] A. Luccioni and J. Viviano, “What’s in the box? an analysis of undesirable content in the Common Crawl corpus,” in *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*. Online: Association for Computational Linguistics, Aug. 2021, pp. 182–189. [Online]. Available: https://aclanthology.org/2021.acl-short.24

[29] E. Dinan, A. Fan, A. Williams, J. Urbanek, D. Kiela, and J. Weston, “Queens are powerful too: Mitigating gender bias in dialogue generation,” in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online: Association for Computational Linguistics, Nov. 2020, pp. 8173–8188. [Online]. Available: https://aclanthology.org/2020.emnlp-main.656

[30] J. Urbanek, A. Fan, S. Karamcheti, S. Jain, S. Humeau, E. Dinan, T. Rocktäschel, D. Kiela, A. Szlam, and J. Weston, “Learning to speak and act in a fantasy text adventure game,” 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)*, 2019, pp. 673–683.

[31] N. Baker Gillis, “Sexism in the judiciary: The importance of bias definition in NLP and in our courts,” in *Proceedings of the 3rd Workshop on Gender Bias in Natural Language Processing*. Online: Association for Computational Linguistics, Aug. 2021, pp. 45–54. [Online]. Available: https://aclanthology.org/2021.gebnlp-1.6

[32] C. Y. Park, X. Yan, A. Field, and Y. Tsvetkov, “Multilingual contextual affective analysis of lgbt people portrayals in wikipedia,” in *Proceedings of the International AAAI Conference on Web and Social Media*, vol. 15, 2021, pp. 479–490.

[33] S. Touileb, L. Øvrelid, and E. Velldal, “Gender and sentiment, critics and authors: a dataset of norwegian book reviews,” in *Proceedings of the Second Workshop on Gender Bias in Natural Language Processing*, 2020, pp. 125–138.

[34] F. Poletto, V. Basile, M. Sanguinetti, C. Bosco, and V. Patti, “Resources and benchmark corpora for hate speech detection: a systematic review,” *Lang. Resour. Evaluation*, vol. 55, pp. 477–523, 2021.

[35] J. Kurrek, H. M. Saleem, and D. Ruths, “Towards a comprehensive taxonomy and large-scale annotated corpus for online slur usage,” in *Proceedings of the Fourth Workshop on Online Abuse*

and Harms. Online: Association for Computational Linguistics, Nov. 2020, pp. 138–149. [Online]. Available: https://aclanthology.org/2020.alw-1.17

[36] P. Badjatiya, S. Gupta, M. Gupta, and V. Varma, "Deep learning for hate speech detection in tweets," in *Proceedings of the 26th international conference on World Wide Web companion*, 2017, pp. 759–760.

[37] T. Davidson, D. Warmsley, M. Macy, and I. Weber, "Automated hate speech detection and the problem of offensive language," in *Proceedings of the International AAAI Conference on Web and Social Media*, vol. 11, no. 1, 2017.

[38] N. Djuric, J. Zhou, R. Morris, M. Grbovic, V. Radosavljevic, and N. Bhamidipati, "Hate speech detection with comment embeddings," in *Proceedings of the 24th international conference on world wide web*, 2015, pp. 29–30.

[39] J. Golbeck, Z. Ashktorab, R. O. Banjo, A. Berlinger, S. Bhagwan, C. Buntain, P. Cheakalos, A. A. Geller, R. K. Gnanasekaran, R. R. Gunasekaran *et al.*, "A large labeled corpus for online harassment research," in *Proceedings of the 2017 ACM on web science conference*, 2017, pp. 229–233.

[40] E. Fersini, P. Rosso, and M. Anzovino, "Overview of the task on automatic misogyny identification at ibereval 2018." *IberEval@ SEPLN*, vol. 2150, pp. 214–228, 2018.

[41] A. Jha and R. Mamidi, "When does a compliment become sexist? analysis and classification of ambivalent sexism using twitter data," in *Proceedings of the second workshop on NLP and computational social science*, 2017, pp. 7–16.

[42] A. Suvarna and G. Bhalla, "# notawhore! a computational linguistic perspective of rape culture and victimization on social media," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, 2020, pp. 328–335.

[43] Y.-L. Chung, E. Kuzmenko, S. S. Tekiroglu, and M. Guerini, "CONAN - COunter NArratives through nichesourcing: a multilingual dataset of responses to fight online hate speech," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, Jul. 2019, pp. 2819–2829. [Online]. Available: https://aclanthology.org/P19-1271

[44] Z. Waseem and D. Hovy, "Hateful symbols or hateful people? predictive features for hate speech detection on twitter," in *Proceedings of the NAACL student research workshop*, 2016, pp. 88–93.

[45] M. M. Manerba and S. Tonelli, "Fine-grained fairness analysis of abusive language detection systems with CheckList," in *Proceedings of the 5th Workshop on Online Abuse and Harms*

(WOAH 2021). Online: Association for Computational Linguistics, Aug. 2021, pp. 81–91. [Online]. Available: https://aclanthology.org/2021.woah-1.9

[46] M. T. Ribeiro, T. Wu, C. Guestrin, and S. Singh, "Beyond accuracy: Behavioral testing of NLP models with CheckList," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, Jul. 2020, pp. 4902–4912. [Online]. Available: https://aclanthology.org/2020.acl-main.442

[47] P. Rottger, B. Vidgen, D. Nguyen, Z. Waseem, H. Margetts, and J. Pierrehumbert, "HateCheck: Functional tests for hate speech detection models," 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)*. Online: Association for Computational Linguistics, Aug. 2021, pp. 41–58. [Online]. Available: https://aclanthology.org/2021.acl-long.4

[48] B. Vidgen, A. Harris, D. Nguyen, R. Tromble, S. Hale, and H. Margetts, "Challenges and frontiers in abusive content detection," in *Proceedings of the Third Workshop on Abusive Language Online*. Florence, Italy: Association for Computational Linguistics, Aug. 2019, pp. 80–93. [Online]. Available: https://aclanthology.org/W19-3509

[49] P. Fortuna, V. Cortez, M. Sozinho Ramalho, and L. Pérez-Mayos, "MIN_PT: An European Portuguese lexicon for minorities related terms," in *Proceedings of the 5th Workshop on Online Abuse and Harms (WOAH 2021)*. Online: Association for Computational Linguistics, Aug. 2021, pp. 76–80. [Online]. Available: https://aclanthology.org/2021.woah-1.8

[50] P. Zhou, W. Shi, J. Zhao, K.-H. Huang, M. Chen, R. Cotterell, and K.-W. Chang, "Examining gender bias in languages with grammatical gender," 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)*. Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 5276–5284. [Online]. Available: https://aclanthology.org/D19-1531

[51] M. Sap, D. Card, S. Gabriel, Y. Choi, and N. A. Smith, "The risk of racial bias in hate speech detection," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics, Jul. 2019, pp. 1668–1678. [Online]. Available: https://aclanthology.org/P19-1163

[52] E. Excell and N. Al Moubayed, "Towards equal gender representation in the annotations of toxic language detection," in *Proceedings of the 3rd Workshop on Gender Bias in Natural Language Processing*. Online: Association for Computational Linguistics, Aug. 2021, pp. 55–65. [Online]. Available: https://aclanthology.org/2021.gebnlp-1.7

[53] N. Nangia, C. Vania, R. Bhalerao, and S. R. Bowman, "CrowS-pairs: A challenge dataset for measuring social biases in masked language models," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online: Association for Computational Linguistics, Nov. 2020, pp. 1953–1967. [Online]. Available: https://aclanthology.org/2020.emnlp-main.154

[54] N. Ousidhoum, Z. Lin, H. Zhang, Y. Song, and D.-Y. Yeung, "Multilingual and multi-aspect hate speech analysis," 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)*. Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 4675–4684. [Online]. Available: https://aclanthology.org/D19-1474

[55] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[56] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 2017, pp. 6000–6010.

[57] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[58] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.

[59] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," *arXiv preprint arXiv:1802.05365*, 2018.

[60] J. Devlin, M.-W. Chang, K. Lee, and K. 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)*. Minneapolis, Minnesota: Association for Computational Linguistics, Jun. 2019, pp. 4171–4186. [Online]. Available: https://aclanthology.org/N19-1423

[61] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019.

[62] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter," *arXiv preprint arXiv:1910.01108*, 2019.

[63] I. Dias, “Customer Support with Sentiment (Masters Thesis),” 2020.

[64] M. L. McHugh, “Interrater reliability: the kappa statistic,” *Biochemia medica*, vol. 22, no. 3, pp. 276–282, 2012.

A

Precision, Recall, and F1-score for Model Training

This appendix contains the precision, recall, and F1-scores of some of the trained models described in Chapter 6, which we were unable to display in the main body of text due to readability concerns. Table A.1 and Table A.2 show these values for the top three experiments of Multi-C and Multi-D, while Table A.3 and Table A.4 do the same for the experiments conducted for NoAge-C and NoAge-D.

	b_none			gender			race			profession			religion			disability		
	P	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
C-E1	0.8275	0.8581	0.8425	0.6106	0.5661	0.5875	0.9581	0.3045	0.4621	0.9694	0.2727	0.4257	0.9075	0.2769	0.4243	0.9295	0.3540	0.5128
C-E2	0.8298	0.8620	0.8456	0.6191	0.5762	0.5969	0.9787	0.0596	0.1124	0.6020	0.6507	0.6254	0.6375	0.6220	0.6296	0.7577	0.6099	0.6758
C-E3	0.8151	0.8770	0.8449	0.5149	0.5946	0.5519	0.6204	0.6591	0.6392	0.6701	0.6043	0.6355	0.5625	0.6285	0.5937	0.6476	0.6255	0.6364
D-E1	0.8225	0.8536	0.8377	0.7515	0.7402	0.7458	0.6942	0.6325	0.6619	0.8755	0.5167	0.6499	0.6736	0.6466	0.6598	0.6960	0.5663	0.6245
D-E2	0.8296	0.8469	0.8382	0.7873	0.7075	0.7453	0.6498	0.6425	0.6461	0.7132	0.5510	0.6217	0.6276	0.6742	0.6501	0.7225	0.5467	0.6224
D-E3	0.8291	0.8454	0.8372	0.7355	0.7510	0.7432	0.6973	0.6289	0.6613	0.8679	0.5134	0.6452	0.6653	0.6530	0.6591	0.6960	0.5830	0.6345

Table A.1: Precision, Recall, and F1-scores for the top 3 experiments for Multi-C and Multi-D, for target categories b_none, gender, race, profession, religion, and disability

	sexual_orientation			gender_identity			nationality			age			non-biased			Overall		
	P	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
C-E1	0.7036	0.5710	0.6304	0.5714	0.6400	0.6038	0.4556	0.5325	0.4910	0.0000	0.0000	0.0000	0.8682	0.8912	0.8796	0.6314	0.5860	0.6046
C-E2	0.6500	0.5909	0.6190	0.5306	0.6797	0.5960	0.4667	0.5385	0.5000	0.0000	0.0000	0.0000	0.8716	0.8913	0.8814	0.6023	0.6059	0.6020
C-E3	0.7107	0.5590	0.6258	0.5612	0.6471	0.6011	0.5000	0.5294	0.5143	0.0000	0.0000	0.0000	0.9006	0.8724	0.8863	0.5912	0.5997	0.5935
D-E1	0.6656	0.6404	0.6527	0.5031	0.5882	0.5424	0.5000	0.4878	0.4938	0.0000	0.0000	0.0000	0.8524	0.9031	0.8770	0.6395	0.5978	0.6132
D-E2	0.6820	0.6136	0.6460	0.6478	0.4858	0.5553	0.6250	0.4505	0.5236	0.0000	0.0000	0.0000	0.8668	0.8937	0.8801	0.6501	0.5830	0.6117
D-E3	0.6557	0.6452	0.6504	0.4843	0.5878	0.5310	0.4750	0.4935	0.4841	0.0000	0.0000	0.0000	0.8506	0.9047	0.8768	0.6324	0.6005	0.6112

Table A.2: CPrecision, Recall, and F1-scores for the top 3 experiments for Multi-C and Multi-D, for target categories sexual_orientation, gender_identity, nationality, age, non-biased, and for the system overall

	b_none			gender			race			profession			religion			disability		
	P	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
C-E8	0.8274	0.8619	0.8443	0.5255	0.6054	0.5626	0.5635	0.6979	0.6235	0.6259	0.6571	0.6411	0.5525	0.6928	0.6147	0.6035	0.6650	0.6328
C-E9	0.8368	0.8461	0.8414	0.5319	0.6068	0.5669	0.5691	0.6941	0.6254	0.6769	0.5994	0.6358	0.5975	0.6530	0.6240	0.6696	0.6255	0.6468
C-E10	0.7965	0.8967	0.8436	0.6511	0.5615	0.6030	0.6442	0.6462	0.6452	0.7041	0.6124	0.6551	0.6800	0.6385	0.6586	0.6960	0.6695	0.6825
C-E11	0.8090	0.8899	0.8475	0.7149	0.5250	0.6054	0.6735	0.6365	0.6545	0.7687	0.5622	0.6494	0.7300	0.5971	0.6569	0.7577	0.6394	0.6935
D-E8	0.8010	0.8739	0.8359	0,6982	0.7750	0.7346	0,6798	0.6301	0.6540	0,5849	0.6275	0.6055	0,5774	0.7077	0.6359	0,6916	0.5836	0.6331
D-E9	0.8086	0.8693	0.8379	0,6845	0.7850	0.7313	0,7036	0.6192	0.6587	0,6830	0.5710	0.6220	0,5649	0.7200	0.6331	0,7225	0.5734	0.6394
D-E10	0.8209	0.8448	0.8327	0.7698	0.7128	0.7402	0.6998	0.6125	0.6532	0.6755	0.6007	0.6359	0.7134	0.5920	0.6471	0.7181	0.5842	0.6443
D-E11	0.8360	0.8256	0.8307	0.7805	0.7106	0.7439	0.6273	0.6492	0.6380	0.7170	0.5740	0.6376	0.6778	0.6304	0.6532	0.7621	0.5545	0.6419

Table A.3: Precision, Recall, and F1-scores for the top 3 experiments for NoAge-C and NoAge-D, for target categories b_none, gender, race, profession, religion, and disability

	sexual_orientation			gender_identity			nationality			non-biased			Overall		
	P	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
C-E8	0.5964	0.5739	0.5849	0.4388	0.7350	0.5495	0.3222	0.5273	0.4000	0.8909	0.8842	0.8876	0.5947	0.6901	0.6341
C-E9	0.6286	0.5847	0.6059	0.5000	0.7050	0.5851	0.4000	0.5217	0.4528	0.8942	0.8777	0.8859	0.6305	0.6714	0.6470
C-E10	0.6464	0.5954	0.6199	0.3163	0.6596	0.4276	0.4444	0.6349	0.5229	0.8968	0.8757	0.8861	0.6476	0.6790	0.6544
C-E11	0.7357	0.5523	0.6309	0.5051	0.6689	0.5756	0.5778	0.5652	0.5714	0.8864	0.8830	0.8847	0.7159	0.6520	0.6770
D-E8	0.5541	0.6654	0.6047	0.4465	0.5820	0.5053	0.5500	0.5000	0.5238	0.8906	0.8816	0.8861	0.6474	0.6827	0.6619
D-E9	0.6033	0.6548	0.6280	0.5283	0.5874	0.5563	0.6375	0.4766	0.5455	0.8937	0.8791	0.8864	0.6830	0.6736	0.6738
D-E10	0.6754	0.6398	0.6571	0.5597	0.5115	0.5345	0.5000	0.5128	0.5063	0.8589	0.8950	0.8766	0.6992	0.6506	0.6728
D-E11	0.7115	0.6218	0.6636	0.5660	0.5233	0.5438	0.6375	0.5152	0.5698	0.8650	0.8901	0.8774	0.7181	0.6495	0.6800

Table A.4: Precision, Recall, and F1-scores for the top 3 experiments for NoAge-C and NoAge-D, for target categories sexual_orientation, gender_identity, nationality, non-biased, and the system overall

B

Annotation Guide for Biased Subtitles

This appendix contains the Annotation Guide that was given to our annotators in the task described in Chapter 6. While the content of the Annotation Guide is provided here verbatim, the formatting has been changed for the sake of readability.

B.1 Introduction

The following .txt file serves as an annotation guide for the task of Bias Classification. Each annotator will be asked to review a total of 400 entries from a subtitle based corpus, evenly divided across 8 decades (50 * 8).

The information is distributed in an excel file. Annotators only know the content of their own file and of the entries they have been assigned

The first part of this document, LOGISTICS, explains the function of the leftmost columns of each file, which are already filled in. The second part of this document, REVIEW, explains the function of the rightmost columns of each file, which are to be filled in by the annotators

Some of the sentences might seem incomplete, incorrect, or otherwise strange. They are directly

obtained from the original corpus, and were not altered by the researchers. Annotators are asked to review the entries as they are, regardless of oddities

B.2 Logistics

- NUMBER: Contains the line number of the csv file that was classified by our model
- FILE_ID: Contains the id of the subtitle file that the entry comes from
- FILE_NUMBER: Contains the number id of the entry in the original subtitle file
- TEXT: Contains the text under review

B.3 Review

The possible values of ”LABEL”, ”TYPE”, and ”CATEGORY” are listed in each sheet.

B.3.1 Label

Takes values “biased” and “non-biased”. “Biased” entries are those which exhibit one or more of the following characteristics:

- Type 1: The presence in the text of slurs or other derogatory terms related to the protected categories. For example, “bitch”, “tranny”, “jew”, or “dyke” are all derogatory terms related to the categories “Gender”, “Gender Identity”, “Religion”, and “Sexual Orientation”, resp.;
- Type 2: The presence of stereotypes or caricatures concerning the protected categories. For example, “You’re pretty smart for a girl”, or “You’re Asian, you must be good at math” refer to stereotypes related to “Gender” and “Race”, respectively;
- Type 3: The presence of otherwise harmful, aggressive, offensive, or derogatory content aimed at people or groups related to the protected categories.

“Non-Biased” entries are entries which don’t feature any of the protected categories. The list of protected categories is in section CATEGORY.

B.3.2 Type

Takes values 1, 2, 3, or 0.

0, if the entry is “non-biased” and 1,2, and/or 3, if the entry is “biased”. The numbers directly correspond to the types of bias described in the previous section.

Annotators may not mark an entry with more than one type in this section. If more than one type seems applicable, please include that information in section OBS. For example, “All women are stupid bitches!” would be, at least, type 1 and 3. In this case, choose the main type in TYPE and add the secondary type in OBS, by hand.

B.3.3 Category

Takes values “Gender”, “Race”, “Profession”, “Religion”, “Disability”, “Sexual Orientation”, “Gender Identity”, “Nationality”, “Age”, or “None”. “None” should be used if the entry is marked as ”non-biased” or if the annotator is unsure of which category the entry is related to.

Annotators may not mark an entry with more than one category in this section. If more than one category seems applicable, please include that information in section OBS.

B.3.4 Obs

This column is meant for observations or notes annotators might find relevant to include. Free format.

B.4 Example Sentences

- “That’s the Chinese” - non-biased; 0; none
- “Come on, he’s an asshole.” - non-biased; 0; none
- “It’s no job for a woman.” - biased; 2 and/or 3; gender
- “And I was like ”I burnt down your house by accident, but you’re a devious slut on purpose.” - biased; 1; gender
- “She said I was a faggot!” - biased; 1; sexual orientation