Generalization of Cyberbullying Traces

Mémoire

Marc-André Larochelle

Maîtrise en informatique - avec mémoire
Maître ès sciences (M. Sc.)

Québec, Canada

Generalization of Cyberbullying Traces

Mémoire

Marc-André Larochelle

Sous la direction de:

Richard Khoury, directeur de recherche

Résumé

De nos jours, la cyberintimidation est un problème courant dans les communautés en ligne. Filtrer automatiquement ces messages de cyberintimidation des conversations en ligne c'est avéré être un défi qui a mené à la création de plusieurs ensembles de données, dont plusieurs disponibles comme ressources pour l'entraînement de classificateurs. Toutefois, sans consensus sur la définition de la cyberintimidation, chacun des ensembles de données se retrouve à documenter différentes formes de comportements. Cela rend difficile la comparaison des performances obtenues par de classificateurs entraînés sur de différents ensembles de données, ou même l'application d'un de ces classificateurs à un autre ensemble de données. Dans ce mémoire, on utilise une variété de ces ensembles de données afin d'explorer les différentes définitions, ainsi que l'impact que cela occasionne sur le langage utilisé. Par la suite, on explore la portabilité d'un classificateur entraîné sur un ensemble de données vers un autre ensemble, nous donnant ainsi une meilleure compréhension de la généralisation des classificateurs. Finalement, on étudie plusieurs architectures d'ensemble de modèles, qui par la combinaison de ces différents classificateurs, nous permet de mieux comprendre les interactions des différentes définitions. Nos résultats montrent qu'il est possible d'obtenir une meilleure généralisation en combinant tous les ensembles de données en un seul ensemble de données plutôt que d'utiliser un ensemble de modèles composé de plusieurs classificateurs, chacun entraîné individuellement sur un ensemble de données différent.

Abstract

Cyberbullying is a common problem in today's ubiquitous online communities. Automatically filtering it out of online conversations has proven a challenge, and the efforts have led to the creation of many different datasets, which are distributed as resources to train classifiers. However, without a consensus for the definition of cyberbullying, each of these datasets ends up documenting a different form of the behavior. This makes it difficult to compare the results of classifiers trained on different datasets, or to apply one such classifier on a different dataset. In this thesis, we will use a variety of these datasets to explore the differences in their definitions of cyberbullying and the impact it has on the language used in the messages. We will then explore the portability of a classifier trained on one dataset to another in order to gain insight on the generalization power of classifiers trained from each of them. Finally, we will study various architectures of ensemble models combining these classifiers in order to understand how they interact with each other. Our results show that by combining all datasets together into a single bigger one, we can achieve a better generalization than by using an ensemble model of individual classifiers trained on each dataset.

Contents

List of Tables

List of Figures

Glossary

Area Under the receiver operating characteristic Curve (AUC) Metric representing the probability of ranking a positive positive sample higher than a random negative sample. 29

Bag-of-Words (BoW) Data structure containing words and their corresponding number of occurrence in a document. 12

Bidirectional Encoder Representations from Transformers (BERT) A popular transformer architecture from the paper [1], differs from the traditional transformer by being bidirectional. 31

Continuous Bag-of-Words (CBoW) Algorithm to learn word vector representations by predicting a word based on the context (surrounding words). ix

DistilBERT A smaller version of BERT trained using distillation, while retaining most of the performance of non-distilled version [2]. 11

distillation Transfer learning technique where a smaller model (student) is trained on the outputs of a bigger model (teacher) in order to reproduce the model's behavior, while having a smaller memory footprint and possibly being faster. viii

dropout A function which randomly masks parts of the input during training to prevent models from overfitting [3]. 30

embedding matrix A matrix of words with their corresponding vector representation. 27

F1-score Corresponds to the harmonic mean of precision and recall, commonly used to express summarize the performance of a model,

$$\frac{precision\ .\ recall}{precision + recall}$$

. 11

FastText FastText is a model based on the skipgram model that includes sub-word information to enrich the learned word vector representations [4]. 23

Global Vector (GloVe) GloVe is an unsupervised learning algorithm for context-based vector representation of words [5]. 9

ngrams Continuous sequence of n characters. 12

precision Proportion of correctly identified positive samples,

$$precision = \frac{TruePositive}{TruePositive + FalsePositive}$$

. viii

recall Proportion of of identified actual positive samples,

$$recall = \frac{TruePositive}{TruePositive + FalseNegative}$$

. viii

Skipgram Algorithm to learn word vector representations by predicting the context (surrounding words) of a word. ix

transformers A neural network architecture popularized by [6] and featuring an encoders-decoders architecture with multiple attention mechanisms. 31

word2vec Group of methods to learn word representation, commonly Continuous Bag-of-Words (CBoW) and Skipgram architectures [7]. 11

[nonumberlist, nopostdot]

Everything should be made as simple as possible, but not simpler.

Albert Einstein

Acknowledgement

Thanks to my research director, Richard Khoury, for his help throughout my master degree and providing me this extraordinary opportunity. Thanks at the entire team at Two Hat, who I had a great time working with, Chris Piebe, Liza Wood, Laurence Brockman, Anne-Marie Thérien-Daniel and Éloi Brassard-Gourdeau, along with every interns I had the pleasure to meet, learn from and work with, Lucas Wu, Charles Poitras, Zeineb Trabelsi and Andre Schorlemmer.

I would also like to thanks all my friends who kept me motivated during my master degree and through harder times.

Introduction

Online social interactions are commonplace nowadays. Unfortunately, with the positive benefits of bringing people together in an unprecedented way, has come the negative impacts, namely the spread of cyberbullying and its social toll. For example, a recent study in Canada [8] found that 17% of 15-to-29-year-old Internet users (or 1.1 million individuals) have experienced cyberbullying, that the problem disproportionately affects women (19%), low-income people (24%), and homosexuals (34%), and that 10% of victims develop emotional, psychological or mental health conditions as a result [9].

However, cyberbullying is rarely commonplace in online communities. For instance, one study of a Wikipedia dataset of 2 million comments found that personal attacks represented only of 0.8% of messages [10]. Consequently, cyberbullying messages can easily hide in the massive amount of content generated by online communities, and it would be impossible for moderators to read and sift through the entire corpus. A possible solution is allowing users report instances of cyberbullying and other toxic behaviors to the attention of moderators. However, constantly reading and reviewing these messages can lead to real health consequences for moderators. In fact, Facebook was recently ordered to pay $52 million to more than 10,000 moderators who developed post-traumatic stress disorder (PTSD) due to their job.[1]

Given the problems associated with human moderation of the communities, many platforms have been looking at the alternative, namely software cyberbullying filters. These filters can vary greatly in sophistication, ranging from simple keyword detection of profanity and obscene words to machine learning algorithms trained on real-world examples of cyberbullying. This however creates a new problem: how can someone determine exactly which messages are and are not instances of cyberbullying in order to create an accurate training corpus? Indeed, many companies and communities have created their own guidelines on what constitutes cyberbullying, and researchers often setup clear definitions to guide their own research, However, there are a lot of variations from one of these definitions to the next, and very little agreement across various definitions on what constitutes cyberbullying. This heterogeneity of definitions has direct negative impacts in the fight on cyberbullying. It means, for instance, that the performance of two cyberbullying detection systems trained on different datasets

[1] https://www.washingtonpost.com/technology/2020/05/12/facebook-content-moderator-ptsd/

cannot be directly compared to each other, and that a system built to detect cyberbullying in one dataset will lack portability and may perform poorly on different datasets.

This thesis makes three key contributions. First, using a thorough literature review, we expose the current state and issues related to the definition of "cyberbullying". Second, we conduct an in-dept analysis of several datasets commonly used in cyberbullying research to train detection systems. Our analysis highlights the traces of cyberbullying present in each dataset, how each dataset compares to the others in terms of vocabulary similarity, and how a cyberbullying detection system trained on each one performs on the others. Finally, our third contribution studies various ways of creating ensemble classifiers that can exploit the full diversity of cyberbullying datasets available.

This thesis is structured as follows. To begin, Chapter 1 describes the variations that exist in the definition of cyberbullying in previous work, along with an in-depth discussion of the current state-of-the-art based on three papers closely related to the subject. Next, Chapter 2 describes in detail each dataset we use in our experiments, followed by an analysis of the similarities between datasets and their vocabulary. Chapter 3 then elaborates on the training framework used for our experiments. Chapter 4 presents our generalization experiment using the previously selected datasets, and chapter 5 presents the results of ensemble models. Finally, we conclude the thesis with some final remarks on our contributions and ideas for future works.

Chapter 1

Related Work

1.1 Variations in the definition

One of the most popular definitions of cyberbullying was proposed by [11], cited more than 3,400 times[1], and reused in several papers working on cyberbullying [12, 13, 14]. It defines cyberbullying as *an aggressive, intentional act carried out by a group or individual, using electronic forms of contact, repeatedly and over time against a victim who cannot easily defend him or herself.* From this definition, we can identify six key components:

1. An aggressive act;
2. With intent to harm the victim;
3. By a group or an individual (not necessarily limited to a single bully);
4. Using electronic forms of contact, which can include email, SMS, discussion board posts, online chat messages, and various other electronic forms of contact;
5. Done repeatedly over time, thus not limited to a single act;
6. Against a victim who cannot easily defend him or herself, indicating a power imbalance between the bully or bullies and the victim.

Although the previous definition is a popular one, many alternatives have been proposed in the literature, by changing or dropping some of the key components. The form of contact and intent to harm components are two that are largely agreed upon. However, while it is straightforward to verify that an electronic form of contact was used, the intent to harm is much more difficult to assess. Indeed, a person's intentions are not easily discernible from their written messages. As a result, evaluating that aspect is often left to discretion of the annotators. This introduces a level of subjectivity in the definition of cyberbullying.

[1] https://scholar.google.ca/scholar?cites=12855620013420392054

An aggressive act is an important component of cyberbullying, but also a vague and ill-defined one. As a result, many authors choose to focus on a specific behavior. While this gives a more clear and precise definition to work with, it does also mean that aggressive acts that fall outside that definition are not considered cyberbullying by these authors. For example, the dataset used by the authors of [15] limits its scope to insulting messages or comments directed towards a specific social media user, and thus excludes aggressive but non-insulting messages as well as insulting messages directed towards someone who is not a user of the platform. Other authors create lists of aggressive acts to look for. An example of this is found in [16], where the set of behaviors used includes flaming (sending rude or vulgar messages), outing (posting private information without consent), harassment (repeatedly sending offensive messages to a single person), exclusion (from an online group), cyberstalking (terrorizing through sending explicitly threatening and intimidating messages), denigration (spreading online gossips), and impersonation (pretending to be another person to mock them). But even such lists cannot be exhaustive. Moreover, they require the introduction of new definitions for each type of behavior, which can lead to ambiguity and conflicts both within the set and with other papers.

The component requiring that cyberbullying be carried out by a group or an individual against a single victim has also given rise to several interpretations. Several papers [17, 18, 15] only consider acts carried out by a single individual towards a singular victim. On the other hand, [19, 20, 14, 21] allow acts carried out by multiple individuals. As for the victim, many authors [19, 13, 22, 14, 21] consider aggressive acts done towards groups instead of a singular victim. Finally, [16] eschews the bully/victim dichotomy and defines a set of roles including people assisting the bully and people defending the victim, making it ambiguous whether a group or individual is bullying or being bullied.

The component of repetition is also one many authors disagree on. This can have the effect of blurring the line between cyberbullying and cyberagression, another problematic online behavior which shares four of the key components of cyberbullying but excludes repetition and the power imbalance [19]. The authors of [19] make this distinction, defining explicitly cyberbullying as an aggressive behavior that is repeatedly carried out and cyberaggression as a single aggressive act. We can point out from this definition that a single act of cyberaggression can snowball into cyberbullying through sharing and liking; however to our best knowledge no current publicly-available dataset features the structure and metadata to allow such detection. On the other hand, many authors have studied datasets of single messages [17, 15, 23, 22], and thus exclude explicitly or implicitly repetition from their definition of cyberbullying. Lastly, the study from the authors of [24] features both single messages and conversations, thus making repetition possible but not necessary to cyberbullying. We should highlight that this question of repetition has consequences on some of the other components of the definition, in particular the type of aggressive act. While some aggressive acts such as insults can be observed from a single message, others, such as harassment or impersonation, requires longer

conversations to become apparent.

The power imbalance component is one the literature is very divided on: many papers omit it entirely from their definition of cyberbullying [17, 25, 22, 21, 24], while an almost equal number of papers include it [19, 16, 13, 20, 14, 18, 15]. However, of those papers that include it in their definition, only [19, 13] actually quantify and measure it based on social media information such as differences in the number of followers and friends of the bully and victim, and [24] uses proxies from user information, such as age and gender, to represent this imbalance. In addition, certain online platforms allow users to post anonymously, therefore making it harder for a victim to easily defend themselves against a bully, which can be considered a form of power imbalance [26]. The authors of [19] show that user information can help predict risks of cyberbullying occurring in posts, indicating that it is an important component of cyberbullying. Despite this, many publicly-available datasets do not have authors associated with their messages at all, and thus make it impossible to measure power imbalance.

This range of variations in the definition of cyberbullying makes it difficult if not impossible to compare the results of different papers with each other. The behaviors detected in each dataset can lead to very different systems. For instance, it is harder to detect outing than direct insults, and a classifier trained for one of these cyberbullying behaviors may not transfer to the other. Moreover, handling and understanding a conversation creates additional challenges compared to understanding a single message, but also makes it possible to detect a larger range of cyberbullying behaviors that simply cannot exist in a single message. Classifiers can improve their results by incorporating repetition or power imbalance but fail in experiments on datasets that do not include this information, while simpler classifiers that beat them on these datasets will in turn give poorer results on the more complete datasets. The multiple issues arising from the various interpretations of cyberbullying found in the literature and in datasets are the key motivation for our work.

1.2 Overview of the current state-of-the-art

In this section, we present three papers related to our project, which also provide an overview of the current state-of-the-art. In the first paper, the authors tackle the task of cyberbullying detection and prediction. In the second paper, the authors demonstrate the non-portability of systems trained from a dataset with one definition of cyberbullying to a different one. Finally, in the third paper, the authors expose the limitations of the current state-of-the-art and propose crowdsourcing content to enrich datasets.

1.2.1 Prediction of Cyberbullying Incidents on the Instagram Social Network

The first paper we study is [19] by H. Hosseinmardi, S. A. Mattson, R. I. Rafiq, R. Han, Q. Lv, and S. Mishr. These authors follow closely the definition of cyberbullying proposed by [11], and include all six components we highlighted. They also define cyberaggression as a single aggressive remark towards a user, and demonstrate that it is a subset of cyberbullying.

First, the authors built a dataset by sampling Instagram media sessions (an initial image and subsequent comments) containing at least 15 comments and one profanity word from a list of profanities. This resulted in 2,218 different media sessions, which were then labelled for cyberbullying or cyberaggression by majority vote of five different annotators. Using quality control methods and test questions inserted during labelling, the authors weighted the annotators' votes via a trust-based metric, which they called the confidence level. Media sessions with a confidence over 60% were kept resulting in a total of 1,954 media sessions. In total, 29% of media sessions were tagged as cyberbullying or cyberaggression, and 71% were clean. Media session were also labelled according to the content of their initial image. For example the labels indicate whether the picture shows a human, text, clothes, tattoos, sports, or celebrities. Multiple labels could be selected for a given image and each image was labelled by three different annotators. The authors highlighted a number of key findings:

1. The annotators generally agree on whether a session contains cyberbullying or cyberaggression. 50% of cyberaggression and 62% of cyberbullying instances were voted as such by at least 4 out of 5 annotators. On the other hand, cases of disagreements where 2 or 3 annotators vote one way and the others vote differently account for only 30% of media sessions.

2. While the cyberbullying and cyberaggression labels were independent, no session received more votes for cyberbullying than for cyberaggression. This seems to indicate that cyberbullying stems from cyberaggression acts.

3. The authors observe that about 30% of medias sessions that were not labelled as cyberbullying nor cyberaggression contained one or multiple profanity words. Consequently, profanity cannot be relied on solely to identify instances of cyberaggression or cyberbullying.

4. The percentage of sessions showing cyberaggression or cyberbullying decreases as the negativity of the messages increases. By examining high-negativity instances, the authors found that while these media sessions do contain a lot of profanity words, they are not insulting to any person in particular. Their analysis has shown that a significantly high level of negativity in a media session correlates with a low probability of it containing cyberbullying incidents.

5. Media sessions that contain cyberbullying have a relatively low comment inter-arrival time compared to non-cyberbullying sessions. These sessions also have a lower number of likes per post while having more followers.

6. Certain labels of image content are highly correlated with cyberbullying. These include *religion*, *death*, *appearance*, *sexuality*, and *drug*. Others have a low correlation, such as is the case for *bike* and *food*.

7. Using only text features and not relying on image content labels, media properties (likes, post time and caption) and user metadata, the authors were able to achieve a precision of 71% and recall of 79% in cyberbullying detection using a Maximum Entropy (MaxEnt) classifier.

8. On the other hand, by using non-text features, such as image content labels, media properties and user metadata, for cyberbullying detection, they achieved a precision of 62% and 76% recall using a MaxEnt classifier.

The observations made in this study are significant. They draw a distinction between cyber-aggression and cyberbullying (which is often overlooked by other authors who tinker with the definition of cyberbullying) and demonstrate that the two phenomena are related (which is why these authors often get interesting cyberbullying detection results despite really studying cyberaggression). They also show that certain topics and language uses are related to cyberbullying, though not cyberbullying themselves. In this thesis we will refer to these as *traces of cyberbullying*, a subset of behaviors associated with cyberbullying but which do not require all the components of cyberbullying, especially repetition or power imbalance, to be identified. Examples of traces of cyberbullying include insults and hate speech.

1.2.2 Deep Learning for Detecting Cyberbullying Across Multiple Social Media Platforms

The authors of [17], S. Agrawal and A. Awekar, highlight three major issues common in papers on cyberbullying. First, these papers often focus on a specific social media platform and cannot be easily generalized to others. Second, they often have a narrow definition of cyberbullying, focusing on one precisely-defined behavior. And third, they rely on manually-crafted features, and thus the models lack flexibility. The authors propose to solve these problems by training a cyberbullying detector using several datasets from different social media platforms. However, we should note that they use a variation of the definition of cyberbullying that ignores repetition, and their dataset is composed only of single messages.

Datasets

The authors have selected three datasets for their study. The first one is from [27], and the source of data is a question-answering platform named Formspring. It features 12,000 annotated question and answer pairs, each manually labelled by three annotators. Of them, 825 were labeled as containing cyberbullying content by at least two annotators.

The second dataset is a Twitter dataset from [28]. It features 16,000 annotated tweets sampled by performing a search of common slurs and terms related to religious, sexual, gender, or ethnic minorities. The dataset features three different classes: 3,117 sexist tweets, 1,937 racist tweets, and the remaining are neither sexist nor racist.

The third and final dataset is from [10] and originates from Wikipedia's talk pages. It contains over 100,000 comments, each labeled by 10 annotators on the presence of personal attacks. 13,590 comments were found to contain personal attacks by majority vote of the annotators.

Each message was truncated to the length of the 95th percentile of the longest post of its dataset. This is done to efficiently train their various models.

The authors first performed an analysis of the usage of swear words and anonymity and its relationship to cyberbullying. FormSpring and Wikipedia both allow for anonymous comments while Twitter does not. This makes it possible to compare the usage of swear words in both anonymous and non-anonymous contexts. The authors do find that swearing is more common in messages from the anonymous platforms. However, they also find that it is not an indicator of cyberbullying. For instance, 82% of cyberbullying tweets do not feature any swear words, while 78% of Formspring posts containing swear words are not cyberbullying. This echoes a similar finding from Section 1.2.1.

Experiments and Results

The first experiment conducted by the authors compares four deep neural networks (DNNs) and four traditional machine learning models. The DNNs are a convolutional neural network (CNN), a long-short term memory network (LSTM), a bidirectional long-short term memory network (BiLSTM) and a BiLSTM with attention. The four traditional machine learning models are a logistic regression (LR), a support vector machine (SVM), a random forest (RF) and naive Bayes (NB). They were trained on two data representation, character n-grams and word n-grams. In this experiment, the traditional machine learning models were found to underperform compared to the DNNs. The best traditional model, the SVM using unigrams, still achieved an F1-score 3% below the DNNs on the Formspring and Wikipedia's datasets.

Their second experiment evaluates the effect of oversampling the minority (cyberbullying) class, a common strategy when dealing with imbalanced datasets. Their experiments show that it does help improve results. However, as [16] points out, the authors performed the

oversampling before the training-testing split, meaning oversampled samples are found in both the training and testing sets. This leads to an over-optimistic performance of the models on the testing set, and their results may not reflect the true potential generalization achieved in the context of unseen data.

Their next experiment contrasts three different initial word embedding representations, namely word embeddings with random initialization, Global Vector (GloVe) context-based word embeddings, and Sentiment-Specific Word Embeddings (SSWE). They found that GloVe and SSWE representations give similar results and both outperform random initialization, though not by a significant margin, on the Twitter and FormSpring datasets.

For their final experiment, the authors evaluated the transfer of the models from one social community to the other. To do this, they trained the DNNs on each one of the datasets and tested it on the others without retraining. Their results show that the models are overfitting to their community, resulting in very low F1-scores when applied in new communities.

To summarize, this paper provides evidence that deep learning models aimed at detecting cyberbullying have trouble generalizing across platforms. It did not however explore in depth why that is. That question is the focus of the next chapter of this thesis.

1.2.3 Current Limitations in Cyberbullying Detection: on Evaluation Criteria, Reproducibility, and Data Scarcity

The authors of [16], C. Emmery, B. Verhoeven, G. D. Pauw, G. Jacobs, C. V. Hee, E. Lefever,B. Desmet,V. Hoste, and W. Daelemans, attempt to address three problems. First, there is a data scarcity problem. Most research in cyberbullying detection is done using small datasets, and as a result the datasets do not represent the language variation found between platforms, both in term of general language use and of bullying specific language. Second, a lot of work focuses on single messages and thus reflect a limited subset of cyberbullying, instead of a more realistic scenario where power imbalance and repetitiveness are taken into account. Finally, they note that not all social media platform allow researchers to collect user profile information and statistics, and many of them do not vet profile information or sometimes even keep detailed user profiles. As a result, a lot of the data needed to detect and model cyberbullying is not present in the datasets. Thus, they hypothesize that crowdsourcing bullying content could alleviate both the influence of domain-specific language-use, while being a solution to data scarcity and allow for richer representations.

For their experiments, they used eight English datasets and four Dutch dataset.

The first and second datasets they used are from Ask.fm, a question-answering network, where users can ask questions on other user's profiles or answer questions on their own profiles,

often while remaining anonymous. One of these datasets is in English and the second one is in Dutch. Both datasets were annotated with fine-grained labels, but for the experiments these labels were binarized. The English dataset features 4,951 cyberbullying and 95,159 safe samples, while the Dutch dataset features 8,055 cyberbullying and 66,328 safe samples.

The next source of data is of cyberbullying messages donated to the Automatic Monitoring in Cyberspace Applications (AMiCA) project by previously bullied teens. It features a mixture of platforms including Skype, Facebook, and Ask.fm. This dataset is relatively small, containing contains 152 cyberbullying and 211 safe samples.

The fourth source is a crowdsourced dataset from an experiment in which 200 teens aged from 14 to 18 participated in a role-playing game on an isolated social network. Each of them was assigned one of the six fictional roles: a bully, a victim, two bystander-assistants, and two bystander-defenders. They had to respond to an artificially generated initial post attributed to one of the group members and were confronted with two initial posts containing low or high severity of cyberbullying. This dataset contains 2,343 cyberbullying and 2,546 safe samples.

The fifth dataset is the same one from [27] used in Section 1.2.2, namely a set of English posts from the question-answering platform Fromspring.me. Each post was labeled by three annotators and a majority vote was then performed. Similarly to the work in 1.2.2, question and answer pairs were merged into a single sample. It features 12,735 samples, 1,024 of which contain cyberbullying.

The sixth dataset is from [29] and was collected from the platform MySpace. It features English messages and was labeled by batches of ten posts, each batch representing one instance with one label. This results in 2,059 samples, 426 featuring cyberbullying content.

The seventh dataset was made available by [30] and was collected from a stream of tweets between 20-10-2012 and 30-12-2012. The tweets were labeled by three annotators and a majority vote was applied. The train set contains 220 cyberbullying and 5,162 safe tweets. The test set was collected by adding a filter to the tweets containing any of the words *school*, *class*, *college* and *campus*. Both sets were merged by the author for their experiments.

The eight dataset also comes from Twitter and was collected by [31]. It focuses on testimonies of cyberbullying events and was retrieved using keywords such as *bully* and *bullying*. This makes is a rather different dataset, as the cyberbullying cases are not actual cyberbullying messages but reports about cyberbullying messages. The dataset contains 4,984 samples, 281 cyberbullying instances and 4,703 safe instances.

As for datasets nine and ten, both are collected from Ask.fm, one featuring English messages the other Dutch messages. However, they were collected at the profile level, and aggregated all messages from each profile into a single instance. If the set of messages contains at least one case of cyberbullying, the aggregated set is labeled as a cyberbulling instance. This

aggregation changes the task towards victim detection. The English dataset contains 1,763 cyberbullying and 6,245 safe instances, but the count for the Dutch dataset is not given.

The next dataset was also collected by [27] on FromSpring.me, however similar to the previous two, it was collected at the profile level. It was built from 49 profiles initially, but to increase the number of instances, if a profile contains more than 5 messages it is split into instances of five messages each. It contains 556 cyberbullying and 756 safe instances.

Finally, the twelfth dataset is a dataset used in a Kaggle competition on toxic comment classification [32]. It contains messages from Wikipedia's talk pages which were annotated for types of toxicity. It contains obscenity, threats, insults and hate speech. The dataset is notably larger than the rest of the datasets and features 15,292 cyberbullying instances of toxicity and 144,274 safe instances.

Experiments and results

The first experiment explores cross-domain evaluation. The authors train a model on each train set and then test it on every test set. In addition one model is trained on the combined train sets. Hyperparameter tuning was done through grid search and using nested cross-validation with then inner and three outer folds. Model selection was done on the outer folds and the best performing model was then re-trained on the full training set (90% of the data) and then applied to the test set (10%). Train-test splits were done in a stratified way, keeping class distribution similar to the corpus. The authors observe a drop of 15-30% in F1-score on models trained on one set and tested on another. Using their baseline, a Linear SVM, they observe that over 75% of the top 5,000 features seen during training do not occur in any test instance and only 3% of features generalize across all sets. Moreover, the coefficient values for the common features can be negative or positive in different sets, meaning they are predictors of cyberbullying in some datasets and of safe messages in others. They conclude that their baseline model does not generalize out-of-domain.

The follow-up experiment attempts to overcome domain influence due to the language use. They conduct three sub-experiments to improve performance: 1) by merging all available training sets in order to simulate a large and diverse corpus; 2) by aggregating instances at the user level; 3) by using state-of-the-art language representations instead of bag-of-words features, namely two pre-trained embedding models per language. For English, they used 200-dimensional GloVe vectors trained on Twitter and DistilBERT sentence embeddings. For Dutch, they used fastText embeddings trained on Wikipedia and word2vec embeddings trained on the COrpora from the Web (COW) corpus[2]. Their results show that the model from the first sub-experiment trained on the combined training sets achieves the best or second-best F1-score all test sets. However, a qualitative analysis of the predictions shows that the model

[2] https://corporafromtheweb.org/

focuses on detecting blatant profanity and that it fails to identify more subtle bullying. Models from the second sub-experiment show a noticeable improvement for in-domain performance, but worse performance on out-of-domain test sets. These models however exploited profile features and did not rely as much on ngrams and profane words, unlike the models of the first sub-experiment, indicating that they could be better at identifying forms of cyberbullying that do not use explicit vulgar language.

Models in the final sub-experiment did not seem to provide any overall improvements, and the authors conclude that no alternative seem to clearly outperform their Bag-of-Words (BoW) baseline.

The third experiment investigates their second hypothesis, that cyberbullying corpora only reflect a limited subset of cyberbullying. They do this by training their BoW baseline on the toxicity detection dataset from Kaggle and testing it out-of-domain on their various English cyberbullying corpora, and vice-versa. The Kaggle corpus defines online toxicity rather than cyberbullying. Their results show that the classifier trained on the Kaggle dataset generalizes better on average than one trained on any individual cyberbullying corpus, but that a classifier trained on the merger of all cyberbullying corpora outperforms any single-corpus classifier in out-of-domain classification. This indicates that cyberbullying content is more complex than what any one dataset represents and models were unable to learn beyond simple features of aggressiveness.

The final experiment proposes a solution to the issues with the current corpora, namely the limited diversity of language in each corpus and narrow definition of cyberbullying found in each corpus, by using crowdsourced data. They focus on the Dutch datasets, and used crowdsourcing to augment these datasets. Their experiments show that even a small amount of data can help improve results using both the Ask.fm and the donated Dutch sets from the AMiCA project. The gain was most notable when using single-message datasets, as opposed to datasets of conversations or with profile information, indicating this may be a way to enrich cyberaggression datasets.

In summary, they found evidence of all three of their hypotheses: previous cyberbullying models do not generalize well across domains and it is difficult to improve above the bag-of-words approach, there is a considerable overlap between toxicity classification and cyberbullying detection, and finally, crowdsourced data yields well-performing cyberbullying detection models.

1.3 Conclusion

In an ideal scenario, cyberbullying detection experiments should be conducted on conversational data, as was the case in the paper in 1.2.1. However, due to the data scarcity problem highlighted by 1.2.3 and the variations and subjective interpretations in the definition of cy-

berbullying itself, it is not always possible to do so, and sometimes it is necessary to use single-line datasets which necessarily omit repetitions. These datasets cannot be said to contain cyberbullying, since a single line cannot fit the definition, but contain rather traces of cyberbullying. While these are more often considered cyberaggression or some form of online toxicity, the authors of [19] have shown that cyberaggression is a subset of cyberbullying, while the experiments of [16] have shown a significant overlap between other forms of toxicity (namely obscenity, threats, insults and hate speech) and cyberbullying. This means that training a classifier to detect certain traces of cyberbullying can be a stepping stone to a more general cyberbullying classifier in the absence of a cyberbullying conversation dataset. This is the key insight that motivates our work. In the next chapters of this thesis, we will study the variability that exists in datasets that contain traces of cyberbullying, experiment with the portability and limitations of cyberbullying classifiers built from them, and develop strategies to create a general cyberbullying detection system that overcomes these limitations.

Chapter 2

Datasets

2.1 Datasets Overview

For our research, we have selected eight datasets pertaining to various types of behaviors that occurring in a repeated way would fit the general definition of cyberbullying. Many of these datasets have labels to designate specific types of behaviors which are either not found in other datasets or defined in a different manner. In order to make the datasets directly comparable, we merged all these labels into a positive "cyberbullying" class, opposing the much more common negative class of messages that do not have traces of cyberbullying. In addition, some of these datasets came already divided into training and testing sets. For the others, we randomly divided the datasets into a 20% test set and 80% training set, which we further divided into 80% training and 20% validation sets. We will describe the datasets in relation to the six components of cyberbullying in table 2.1.

2.1.1 Hate Speech and Offensive Language (dataset A)

This dataset[1] was collected by [33] by searching Twitter for tweets containing hate speech terms from the lexicon *Hatebase.org*. Out of the 85.4 million tweets gathered, 25,000 tweets were randomly selected and annotated by three or more annotators to one of three labels: if it contains hate speech, if it contains offensive language without hate speech, or if it contains neither hate speech nor offensive language. A tweet's final label was the majority decision, and tweets for which no majority decision existed were filtered out. This gave them a corpus of 4,163 tweets that do not contain hate speech nor offensive language, 19,190 that contain offensive language, and 1,430 that are considered hate speech, making it the only corpus in our study imbalanced in favor of the positive class. After we merged the two positive labels and split the dataset into training, validation, and testing corpora, this yields a training corpus of 3,002 negative-class tweets and 14,841 positive-class tweets, making it the only corpus imbalanced in favor of the positive class. It includes direct mentions of people, retweets and

[1] https://github.com/t-davidson/hate-speech-and-offensive-language

hashtags. Here are a few example of hate speech and offensive content found within the dataset:

- *Real niggas speaking bitch u better listen....*
- *@VinniePolitan the cunt needs her hands cut off!!!!!*
- *RT @_LOSOWORLD: @ChiefKeef bitch ass nikka took Flight Frm #Kenosha*

and a few non-offensive and non-hate speech comment within the dataset:

- *I'm a redneck, but without the homophobia and racism*
- *@AdamBaldwin @ChicoDelainky clams to be a comedy writer. #untrue*
- *RT @intelwire: Significant number of US tweets were Americans flipping the bird to IS. UK, not so much.*

2.1.2 Racism and Sexism (dataset B)

The authors of [28] designed a list of slurs and terms identifying religious, sexual, gender, and ethnic minorities. They then sampled Twitter for tweets using these words, and refined their list based on the results. They manually annotated each sampled tweet as sexist, racist, or neither, and had an expert review their annotations in order to mitigate possible bias. We used a version of their dataset[2] made available by [17]. After merging the two positive classes and splitting the dataset, we obtain a training corpus of 7,957 negative and 3,627 positive tweets. Similarly to the previous dataset from Twitter, it includes direct mentions and retweets. Here are examples illustrating the racism and sexist content found within the dataset:

- *oh my god. kat is such a fucking bitch. so much hate #mkr*
- *rt @vsvpgoblin i'm not sexist , but some bitches ca not drive for shit.*
- *@middleeasteye if you want to understand isis , understand islam. they are exactly the same.*

and examples of non-racist non-sexist content within the dataset:

- *if my day does not get completely run off the tracks , i have got a bunch of writing to do and blog posts to schedule. some quick wins.*

[2] https://github.com/sweta20/Detecting-Cyberbullying-Across-SMPs

- *@umarmal if you are too ignorant to know any history , then get lost. if you want to learn , read about charles martel.*
- *rt @finicalgal: basically the #mkr "twists" are a series of annoying ways to extend this season.*

2.1.3 Bullying (dataset C)

This dataset was gathered by the authors of [27] from the question-answering (QA) platform *Formspring*. They randomly selected 50 of the 18,554 users of the platform, each with between 1 and 1,000 QA pairs on the site. Each pair was labeled by three annotators who were asked to judge if it contains cyberbullying and, if so, to identify which words or phrases were the reason. This created a labeled dataset of 12,773 QA pairs[3]. We follow the work of [17] and concatenate the question and answer into as single message. After splitting the dataset, we obtain a training corpus of 8,619 negative and 556 positive samples. Here are a few examples of concatenated question and answer featuring bullying content:

- *why are you friends with that hoe ass bitch daniela ? why do you care soo much hater*
- *ewwwwwww okay i know im ew*
- *u so ugliii ew hoe ghet a lyfe bitch yu so soft yuu get a fuckinn life stop worryinn bowt miness yu fuckk hoe*

and examples of concatenated question and answers without bullying content:

- *in school what clique did do you fit into ? the rednecks*
- *do you crack your knuckles ? oh god no ! that creeps me out ! besides being really bad for your bones it sounds horrible s do you ?*
- *i like kiwis no you love kiwis lmao u said so*

2.1.4 Insults in social commentary (dataset D)

This Kaggle competition dataset [34] was gathered from an unspecified social networking site, and the comments it is comprised of were labeled as insulting or not. However, this dataset applies a narrower definition and only labels positive messages if they are obviously insulting to a specific member of the on-going conversation. Messages that insult celebrities and public figures, messages that include insults and racial slurs not directed at a specific person, and subtle insults, are all counted as negative-class messages. While this dataset did come with

[3] https://www.kaggle.com/swetaagrawal/formspring-data-for-cyberbullying-detection

a separate test set from its creators, we split its training set into a training and validation set and obtain a training corpus of 2,898 negative messages and 1,049 positive messages. On this platform, people sometimes directly refer to someone using the @ sign, however they do so rarely. Here are some examples of the positive class instances featuring insults to another participant of the conversation:

- *shut the fuck up. you and the rest of your faggot friends should be burned at the stake*
- *god, you're tiresome. get a life, you loser.*
- *That you are an idiot who understands neither taxation nor women's health.*

and a few examples of the negative class:

- *Yeah and where are you now?*
- *@sdnoriko LOL as if Capcom listens. Watch them fuck this up too. I hope I am wrong. I love Okami.*
- *you better hope because the Marlins have a history of giving up their superstars*

2.1.5 Hate Speech (dataset E)

This dataset is composed of messages scraped from the white-supremacist internet forum *Stormfront* by the authors of [35]. The messages were labeled into one of four classes. The "hate" class is for messages that are (i) deliberate attack (ii) directed towards a specific group of people (iii) and motivated by aspects of the group's identity. This definition is different from traditional hate speech: for example using a racial slur in an offhand manner and without deliberately attacking that racial group is not considered a hate-class message. The "relation" class is for messages that do not fit in the hate class by themselves, but do when read within the context of the conversation in which they appear. The "skip" class contains non-English messages and gibberish. Finally, the "non-hate" class is for all messages that do not fit the other three categories. The original dataset[4] is composed of 9,507 non-hate messages, 1,196 hate messages, 168 relation messages, and 73 skip messages. We redefined the hate class as our positive class and the others as our negative class. The rationale for including relation in the negative class is that most cyberbullying detection systems work on a message-by-message basis and will mark such messages without context as acceptable. After splitting, we obtain a training corpus of 7,002 negative messages and 877 positive messages. Here are examples of comments featuring hate speech content from the forum's discussions:

[4] https://github.com/aitor-garcia-p/hate-speech-dataset

- *Right now when I think of WASP's , I think of New England liberals who have never had a taste of diversity in their life.*
- *Always nice to see a negro on the receiving end of a beat down from a white for a change.*
- *Boy, those same blacks get awful mad when you have a sign saying "No to a black Africa", then it becomes wrong.*

and comments without hate speech content:

- *Stoning - Monty Python's Life of Brian - YouTube*
- *Blacks and Muslims are the majority.*
- *I just had an extreme mini blizzard (cookie dough) from Dairy Extremist.*

2.1.6 Toxic comment classification (dataset F)

The second Kaggle dataset [32] was gathered from Wikipedia talk page comments. Unlike some of the other datasets, this one features relatively long messages and sometimes entire paragraphs of text. Each comment was annotated with six different labels: toxic, severely toxic, obscene, threat, insult, and identity hate; however the annotation procedure has not been disclosed. Unlike the previous datasets, each comment can be tagged with multiple labels. The dataset contains 21,384 comments labeled as toxic, 1,962 as severely toxic, 12,140 as obscene, 689 as threats, 11,304 as insults, and 2,117 as identity hate. For our work, we assign a comment to our positive class if it contains any one or more of these labels. We also kept the test corpus intact, but split the training corpus into training and validation corpora. After splitting, our training corpus contains 12,979 positive-class and 114,677 negative-class comments. Here are comments from the dataset featuring toxicity:

- *Bye! Don't look, come or think of comming back! Tosser.*
- *You are gay or antisemmitian?*
  Archangel WHite Tiger
  Meow! Greetingshhh!
  Uh, there are two ways, why you do erased my comment about WW2, that holocaust was brutally slaying of Jews and not gaysGypsysSlavsanyone...
  1 - If you are anti-semitian, than shave your head bald and go to the skinhead meetings!
  2 - If you doubt words of the Bible, that homosexuality is a deadly sin, make a pentagram tatoo on your forehead go to the satanistic masses with your gay pals!
  3 - First and last warning, you fucking gay - I won't appreciate if any more nazi shwain would write in my page! I don't wish to talk to you anymore!
  Beware of the Dark Side!

- *Stupid peace of shit stop deleting my stuff asshole go die and fall in a hole go to hell!*

and here are comments labeled as not containing toxicity:

- *Are you threatening me for disputing neutrality? I know in your country it's quite common to bully your way through a discussion and push outcomes you want. But this is not Russia.*
- *Locking this page would also violate WP:NEWBIES. Whether you like it or not, conservatives are Wikipedians too.*
- *April 2006*
  Thank you for experimenting with the page Andy Griffith on Wikipedia. Your test worked, and has been reverted or removed. Please use the sandbox for any other tests you want to do. Take a look at the welcome page if you would like to learn more about contributing to our encyclopedia.

2.1.7 Unintended bias in toxicity classification (dataset G)

This dataset, also from Kaggle [36], was obtained from a news website comment filter system called *Civil Comments*. Its 1,999,514 comments were labeled by three to ten crowd-sourced annotators on average (and sometimes up to a thousand annotators) into six labels: severe toxicity, obscene, threat, insult, identity attack, and sexually explicit. Any and all labels chosen by half or more of annotators was applied to the comment. Once again, we consider a comment as belonging to the positive class if the target threshold is equal or above 0.5. After splitting, we obtain a training corpus of 85,150 in the positive class and 1,358,749 comments in the negative class. Here are a few examples of comments featuring toxicity:

- *haha you guys are a bunch of losers.*
- *ur a sh*tty comment.*
- *It's ridiculous that these guys are being called "protesters". Being armed is a threat of violence, which makes them terrorists.*

and comments labeled as not containing toxicity:

- *The ranchers seem motivated by mostly by greed; no one should have the right to allow their animals destroy public land.*
- *Thank you!! This would make my life a lot less anxiety-inducing. Keep it up, and don't let anyone get in your way!*

- *This is so cool. It's like, 'would you want your mother to read this??' Really great idea, well done!*

2.1.8 Personal attacks and harassment (dataset H)

The final dataset we selected for our study was also constructed from Wikipedia talk page comments. The authors of [10] randomly sampled the 63 million talk-page comments posted between 2004 and 2015. They added comments from a set of users blocked for violating Wikipedia's policy on personal attacks, sampling five comments per user shortly before they were blocked. These were given to a group of annotators, who were asked if each comment contains a personal attack or harassment, whether it is targeted at the recipient or a third party, if it is being reported or quoted, and if it is another kind of attack or harassment. Data quality was assured by requiring that each annotator label ten test comments, and quality control comments were also inserted during the annotation process. Each comment was annotated by ten separate annotators. This resulted in a corpus[5] of 115,859 comments, of which 13,590 were found to contain personal attack or harassment. We used the train and test split made by [17], and further divided their training corpus into a training and validation set. Our training corpus thus contains 6,463 positive-class and 49,155 negative-class comments. Here are personal attacks found within the dataset:

- *i have a dick, its bigger than yours! hahaha*
- *== renault == you sad little bpy for driving a renault clio which has no vaa voom so there and the mcflurry is made of shit*
- *== fuck you thue == you brain dead fuck, you can't block me. thanks to proxy servers and anon browsing i can come to wiki zillion times so if scum like you think that you can block me then you are in for big surprise you faggot. i am giving you last chance in civilized way to stop vandalizing kash jaffrey. don't fuck with canadian content, its to advance for your empty brain to concieve. and lastly if you want to leave message then write (don't use predefine one liners.) see ya asshole joey*

and examples of comments not featuring personal attacks:

- *== please do not abuse vfd process == there is a process of community voting, your lone-rangerism is not appreciated.*
- *sorry that was a typo on my part. i meant to say trademark. .*
- *:it's not 100% clear, but there was a white flash while riding in the osprey. i don't remember if there is dialog in hl2 that would clarify.*

[5] https://doi.org/10.6084/m9.figshare.4054689.v6

Table 2.1: Cyberbullying definition components in each dataset

Dataset	Aggressive Act	Intend to harm	By an individual	By a group	Electronic Forms	Repetition	Power Imbalance
A	Yes	Yes	Yes	Retweets	Yes	Retweets	Implied
B	Yes	Yes	Yes	Retweets	Yes	Retweets	Implied
C	Yes	Yes	Yes	Yes	Yes	Yes	No
D	Yes	Yes	Yes	No	Yes	No	No
E	Yes	Yes	Yes	No	Yes	No	Implied
F	Yes	Yes	Yes	No	Yes	No	No
G	Yes	Yes	Yes	No	Yes	No	Implied
H	Yes	Yes	Yes	No	Yes	No	No

We should note that there is an overlap between the training set of the toxic comment classification competition (dataset F) and test set of this dataset. Both were sampled from the same bigger dataset featuring 6.3M comments from Wikipedia's talk pages. After the competition launched, its organizers realized that some of their test corpus was included in this dataset.[6] They corrected the problem by moving these messages to their training corpus, which solved the issue of getting a high score in the competition through overfitting, but left the overlap in place.

2.2 Dataset Vocabulary Comparison

The eight datasets we selected were collected from platforms with different messaging formats, from limited-length tweets to question-answer pairs to forum conversations. They all pertain to traces of cyberbullying [26], although only [27] explicitly names it as such, and the problematic behaviors they monitor varies greatly, from the focused scope of the Twitter corpora to the wide range of behaviors labeled in the Kaggle datasets. Some behaviors are common; hate speech is explicitly labelled in five of the eight datasets. And some behaviors are unique to some corpora; threats are explicitly noted in only two corpora and sexually-explicit comments in one. This wide variety of traces of cyberbullying and its consequences are the main focus of our study.

To begin, we will study the impact of this diversity on the vocabulary of the datasets using both traditional cosine similarity and word embeddings. For this study, we divide each dataset into its positive and negative portions and treat each as a separate dataset. Basic statistics are given in Table 2.2. In this table, the dataset letters refer to those in the the subsection headings of Section 2, and the subscript + and - to the positive and negative class.

To compute the cosine similarity, we convert each dataset into its bag-of-word representation and compute the TFIDF value of each word using the standard formula:

$$tfidf(w,d) = (1 + \log n_{w,d}) \times \log \frac{D}{D_w}, \tag{2.1}$$

[6] https://www.kaggle.com/c/jigsaw-toxic-comment-classification-challenge/discussion/46177

Table 2.2: Number of messages, unique words, and total word count in each dataset

Dataset	Messages	Unique words	Total words
A_-	3,002	10,989	34,606
A_+	14,841	23,193	155,359
B_-	7,957	15,153	73,973
B_+	3,627	9,531	40,027
C_-	8,619	14,445	104,946
C_+	556	2,554	8,049
D_-	2,898	13,195	56,848
D_+	1,049	4,377	13,183
E_-	7,002	12,468	61,023
E_+	877	3,706	10,273
F_-	114,677	162,524	4,410,757
F_+	12,979	32,001	398,342
G_-	1,358,749	284,164	38,232,678
G_+	85,150	63,441	2,032,318
H_-	49,155	99,083	1,983,200
H_+	6,463	21,748	235,820

where the TFIDF value of word w in dataset d is the log normalization of the number of times the word occurs in the dataset ($n_{w,d}$) times the inverse log of the number of datasets D (16 since we handle the positive and negative class of each dataset separately here) and D_w the number of datasets containing word w. Using the log normalization of the word count (defined as 0 if a word does not occur in a corpus) instead of using the word count directly helps mitigate the impact of the extreme difference in the size of our datasets shown in Table 2.2, by focusing on the order of magnitude of the counts instead of their values. Once each dataset is represented by its TFIDF-weighted word vector, we compute the cosine similarity between each pair of dataset:

$$\cos(\mathbf{v}, \mathbf{w}) = \frac{\mathbf{v} \cdot \mathbf{w}}{\|\mathbf{v}\| \cdot \|\mathbf{w}\|}, \tag{2.2}$$

where $\mathbf{v}$ and $\mathbf{w}$ are the word vectors of two datasets. The cosine similarity is a vector distance metric in the vocabulary space, ranging from 0 for two completely different vectors to 1 for two identical vectors. Measuring this for each pair of our 16 classes gives the results presented in Table 2.3. These results show that, for 6 of the 8 datasets, the most similar dataset to its positive class is its negative class and vice-versa. The two exceptions are datasets F and H, where the negative and positive class of each is twice as similar to the negative and positive class of the other than to its own complementary class. Since both of these datasets were constructed from Wikipedia talk page comments, this similarity is not surprising. The similarity between the positive and negative class of dataset G is a lot higher

than the similarity between the classes of other datasets, but this can be explained by the fact this dataset comes from news comments and so its conversations were constrained to news topics and vocabulary, unlike the other corpora in which users could discuss any topic at all.

In fact, what is surprising in Table 2.2 is not the similarities we find, but how few of them we find. Datasets A, B, C, D and E have nearly no similarity to any other dataset. Datasets A and B, both from Twitter, have no more similarity to each other than they do to datasets from other sources. Likewise, the positive classes of datasets A and E, both focusing on hate speech, have no more in common than they do to other datasets, positive or negative. Even the similarity between the positive and negative class of each dataset is rather low. Normally we would expect language to be homogeneous throughout a dataset and to vary mainly on the presence or absence of class-specific cyberbullying keywords, and since those would be a minority of the words used the similarity between the positive and negative classes should be high. The low values observed indicate the opposite, that the positive and negative classes of each dataset differ also on the non-cyberbullying vocabulary used.

Finally, the low similarity between the positive classes of different datasets shows that the vocabulary marking cyberbullying is very different from one dataset to the next. This can be attributed in part to the different cyberbullying behaviors each dataset measures and to the wildly different platforms each dataset was collected from, and of course to the diversity and flexibility of the English language. The consequence, however, is that we should expect a system trained to recognize cyberbullying in one dataset to have a lot of difficulty picking out cyberbullying in another.

Table 2.3: Cosine Similarity between each pair of datasets

	A_-	A_+	B_-	B_+	C_-	C_+	D_-	D_+	E_-	E_+	F_-	F_+	G_-	G_+	H_-	H_+
A_-	1.000															
A_+	0.212	1.000														
B_-	0.013	0.010	1.000													
B_+	0.009	0.007	0.326	1.000												
C_-	0.014	0.027	0.011	0.006	1.000											
C_+	0.005	0.016	0.003	0.002	0.188	1.000										
D_-	0.014	0.013	0.011	0.009	0.013	0.004	1.000									
D_+	0.013	0.013	0.009	0.007	0.008	0.005	0.249	1.000								
E_-	0.013	0.012	0.015	0.011	0.016	0.003	0.024	0.013	1.000							
E_+	0.009	0.010	0.010	0.008	0.008	0.002	0.015	0.016	0.094	1.000						
F_-	0.026	0.024	0.032	0.021	0.042	0.010	0.048	0.023	0.074	0.032	1.000					
F_+	0.024	0.029	0.033	0.024	0.041	0.018	0.048	0.037	0.066	0.038	0.274	1.000				
G_-	0.030	0.030	0.035	0.022	0.048	0.012	0.052	0.025	0.067	0.032	0.284	0.151	1.000			
G_+	0.036	0.036	0.046	0.034	0.051	0.013	0.074	0.042	0.085	0.049	0.276	0.206	0.545	1.000		
H_-	0.026	0.024	0.036	0.024	0.042	0.010	0.051	0.024	0.083	0.038	0.519	0.301	0.258	0.283	1.000	
H_+	0.021	0.024	0.031	0.022	0.035	0.015	0.042	0.032	0.062	0.036	0.228	0.510	0.126	0.182	0.303	1.000

Next, we select the 30 most important words of each dataset class according to the TFIDF value. The complete list of these words can be found in Appendix B. We get the word embedding representation of each word using FastText. While cosine similarity considers ex-

act word matches and word counts, word embeddings highlights semantic similarity between different words. We project these 300-dimensions word embeddings onto a 2D map using the t-distributed stochastic neighbor embedding (t-SNE) [37], a nonlinear dimensionality reduction technique which projects high-dimensional objects onto a plane by mapping similar vectors to nearby points and dissimilar vectors to distant points. This results in the graphical representation of Figure 2.1. The decision to limit ourselves to 30 words per class is simply to avoid overcrowding the graphic. In this figure, each color represents one of the 8 datasets, and the circles and triangles represent the negative and positive class.

This representation of the 480 most significant words of our datasets illustrates the vocabulary diversity problem mentioned earlier. We do observe some small homogeneous regions, mostly on the edges of the figure, where positive or negative words of several datasets cluster near each other. These regions represent words of similar meaning labeled in the same class in different datasets. Ideally we would want the entire graphic to have regions for words of clearly safe meaning and for words of clearly cyberbullying intent agreed upon by all datasets. However, most of the graphic is a large heterogeneous zone, with words of similar meanings labeled in the positive class in one dataset and the negative class in another. These are neutral-meaning words that appear in only one class of a dataset, and thus become significant indicators of that class. Figure 2.1 shows that this is very frequently the case. Combined with our previous observation on cosine similarity, this further highlights the difficulty of transferring a cyberbullying detector trained on one dataset to another.

2.3 Conclusion

In this section, we selected eight datasets commonly used for cyberbullying research and began studying them. The first stages of our research consisted in comparing their definitions of cyberbullying and their vocabulary use. We found a huge diversity on both counts. Indeed, not only is there little agreement on how to define cyberbullying and very little vocabulary in common between datasets, but words with similar meanings are labeled in contradictory ways in different datasets. This explains the findings in 1.2.3 and 1.2.2, where both papers found that cyberbullying detection systems trained on one dataset had limited portability, and also quantifies the problem of data sparseness mentioned in 1.2.3.

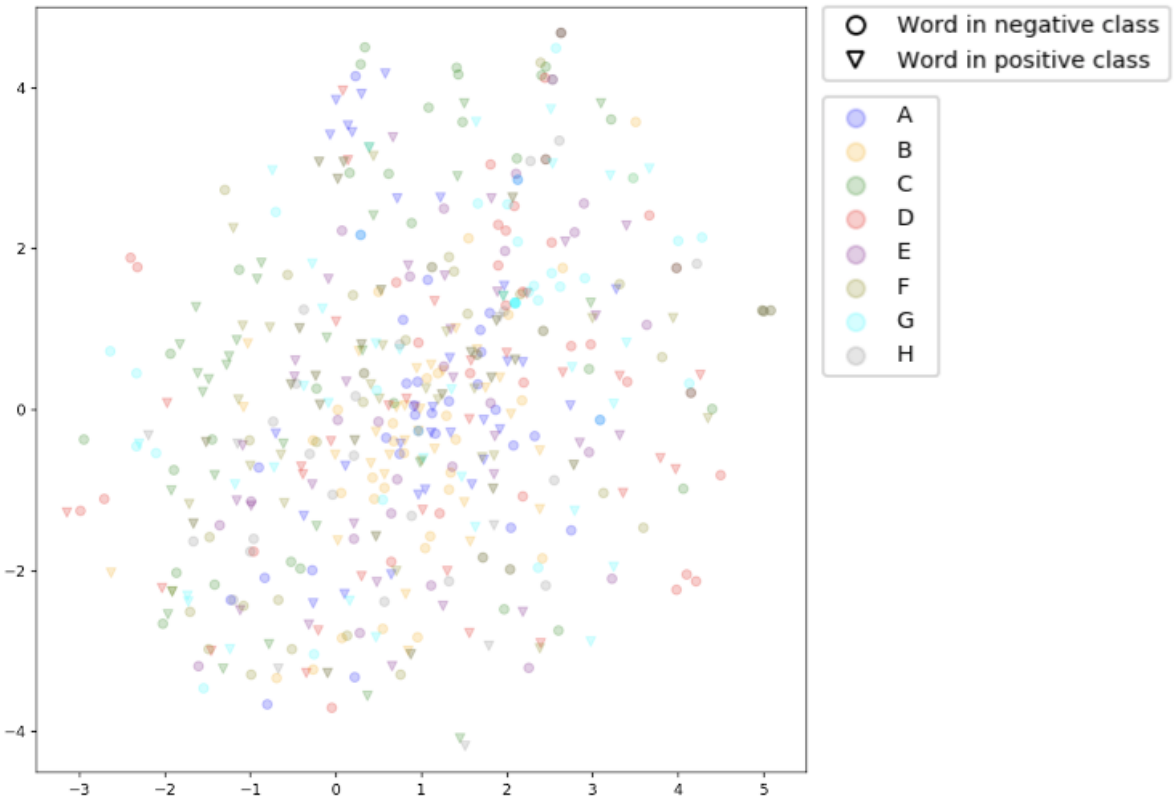

Figure 2.1: t-SNE projection of the top 30 words of the positive and negative class of each dataset.

Chapter 3

Training Framework

In order to conduct our experiments of the next two chapters, we developed a training framework with consumer-grade hardware in mind. Every experiment we will present was conducted using a Ryzen 2700X CPU paired with a Nvidia RTX 2070 (8GB) along with 48GB of RAM. It was developed mainly using Pytorch 1.3.[1] Additionally, this setup allowed us to debug and iterate faster, while being able to consistently reproduce our results.

3.1 Reproducibility Features

The most important requirement when developing a training framework is reproducibility. This is generally achieved by seeding the random number generator. Additionally, although not all CUDA functions can be turned into deterministic functions,[2] we made configurable the use of the stricter setting in Pytorch's library that allows the use of deterministic functions where possible when running the models. However, we found no significant difference in the results between runs on different hardware with and without that setting, and given its cost in computation time we do not feel it is worth using.

3.2 Preprocessing, Portability and Modularity

Before we can run any algorithm, we must prepare the data with a preprocessing stage. We opt for a modular approach using a composite design pattern, a design pattern popularized by E.Gamma, R.Helm, R.Johnson, and J.M. Vlissides in [38], which allows the use of a hierarchical tree structure and in our context to apply preprocessing rules and feature extraction steps. We have three main types of operations: concatenation, replacement, and addition, all of which can be applied to a sequence or element-wise. The first operation, concatenation, simply concatenates the result at the end of the given element or sequence. The second,

[1] https://pytorch.org/

[2] https://pytorch.org/docs/stable/notes/randomness.html

replacement, replaces an element or a sequence; an element can be replaced by a sequence and vice-versa. Finally, the addition operation is simply a regular addition to an element or sequence. We can add either tensor or a numerical value. The leafs of the tree structure are where the preprocessing or features extraction operations are executed. Figure 3.1 illustrates the preprocessing and feature extraction structure used for our experiments. First, we preprocess the sentence by replacing it by a lowercase version of it, then tokenize the sentence using the NLTK library.[3] Next, feature extraction is done by replacing each word with an encoded representation of that words (element-wise). In our case this is done using a FastText network pre-trained on Common Crawl data featuring 300 dimensions and 2 million word vectors with subword information[4] to convert the words into vector representations. In addition to that word representation, we perform a one-hot character encoding of the words (once again element-wise) using a 60-dimensional binary vector representing 60 common characters; each character appearing in a word is marked as a 1 in this vector. This makes the system more robust to misspellings and typos (which are very common online): a word that is a misspelling of another may be a distant vector in word embedding space, but it will be nearby in character space. This is an idea lifted from [39]. Finally, we concatenate the one-hot encoding of the words to the previous encoding. The modular structure we used allows us to quickly add, remove and/or modify the steps of our preprocessing and feature extraction. For example, future work could easily add a lemmatization step before the concatenation, or part-of-speech tagging during the concatenation step.

3.3 Data Input

Given that we work with multiple datasets each with its own file format, we implemented the adapter design pattern, also popularized by the authors of [38], to extract the data from the files. This acts as an abstraction layer between the input files and our framework. Thus our framework doesn't need to know the details of each dataset's file format, only the structure of the data extracted from them, which in our case is some string of text with a label. We performed a similar abstraction for the encoding of words using embeddings by using an interface that accepts a word and returns a vector representation of it. Thus we can support different embedding such as regular word2vec, Fasttext's binary word embeddings or an embedding matrix. For our experiments we used FastText word embeddings, but future work could try other alternatives for comparison.

3.4 Optimizations

In addition to the typical optimization step of running our models on a GPU, we implemented the loading of our data in memory in a lazy-loading manner using Python's generators. This

[3] https://www.nltk.org/

[4] https://github.com/facebookresearch/fastText

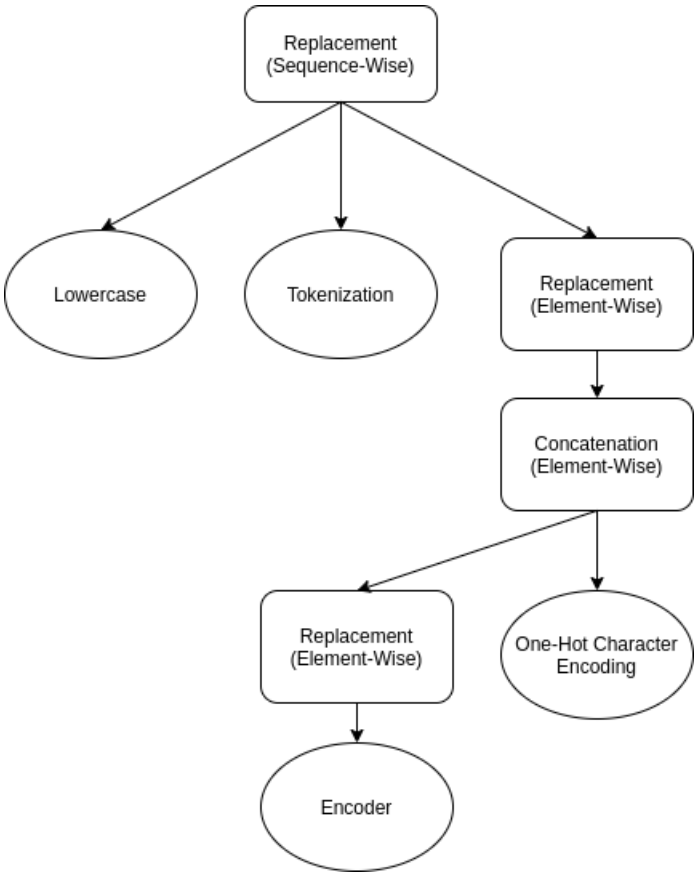

Figure 3.1: Composite design pattern for preprocessing and features extraction.

means that our system only loads the samples it needs in memory instead of the entire dataset. However, due to issues with Python's memory management, it is still possible to have the memory consumption slowly increases when working with text strings using multiple threads; consequently a large enough dataset can easily fill up the computer's available memory. To address this issue, we implemented an efficient way to split and save the dataset to a PostgreSQL database. Given a set of N messages in a dataset that we need to split between training and validation (or test) sets, we generate a list of unique random integers between 0 and N-1 of length equal to the desired size of the validation set. Then, we scan through the list of messages sequentially, and save each message to either the validation database if its index is in our list or to the training database otherwise. This algorithm allows us to split the dataset at minimal performance and memory consumption costs (and especially without loading the entire dataset into memory at once), and allows us to address the messages by index number in the database afterwards. Moreover, we preform this split at the class level (positive or negative) instead of the dataset level, in order to maintain the class distribution of each dataset.

This reduced memory usage allows us to use sequence bucketing, a technique that consists of

padding the messages to match the longest message of a batch (a set of 128 messages in our case) instead of truncating or padding the entire dataset to a fixed length, which would result in either memory waste (if there are a few very long messages) or lost information (if longer messages are truncated). Since this technique doesn't require to find an optimal truncating length, it can easily adapt to new datasets without any changes.

Another optimization comes from the use of Nvidia's Apex library[5], which allow mixed precision training on Nvidia's Pascal architecture graphic cards and more recent ones. It uses less memory and is faster to compute. There is a trade-off in precision, however in our experiments we found it to have almost no negative impact [40].

3.5 Logging and metrics

Finally, we also implemented logging of the important steps, such as splitting, training epochs, and the final results. The logging supports both Jupyter notebooks and writing to an external file. In addition, evaluation metrics to run on the results were implemented in a modular way, and thus can be extended in the future to allow custom metrics. Our framework currently supports accuracy, precision, recall, F1-Score, confusion matrix and the Area Under the receiver operating characteristic Curve (AUC) score. In addition, precision, recall and F1-Score metrics have support for macro, micro and weighted averaging, as macro can be heavily affected by class imbalance depending on the scenario and the desired class importance.

[5] https://github.com/NVIDIA/apex

Chapter 4

Generalization Experiment

For our first set of experiments, we explore the generalization power of a cyberbullying detector trained on any one of the datasets of Chapter 2 to the other datasets. This is an important result to document: while every one of the datasets has been used to train detectors that perform very well on a test corpus that comes from the same source as its training corpus, experiments on other datasets are more rare in the literature, and when done [26, 16, 17] they do not explore the limitations of this transfer in great depth. Moreover, it is a result of immense real-world consequence. Publications in online communities are very heterogeneous in style, content, and acceptability standards, as our sample of datasets has shown. Establishing the portability of a cyberbullying detector trained for one community is thus very important.

4.1 Model and Training

There exists many deep neural network architectures trained for cyberbullying detection. For our experiments, getting state-of-the-art results is not as important as getting comparable results on all datasets that will allow us to contrast their strengths and weaknesses. We thus opted for an attention bidirectional long-short-term model (LSTM). This model features two bidirectional LSTM layers of 128 hidden units each, followed by a scaled-dot-product attention, global max and average pooling layers, and finally three linear layers the size of the concatenated pooling and attention layers. We apply layer normalization on the input, after the two bidirectional LSTM layers, and after the attention layer. In this architecture, the global average pooling summarizes the features in lower dimensions, masking some potential noise, while, on the other hand, the global max pooling can highlight a specific set of features important for the classification. We perform dropout after the initial layer normalization and before the last linear layer. The activation function is the gaussian error linear units (GELUs), which consistently exceeded accuracy on numerous datasets as an alternative to ReLUs or ELU [41]. The output of our model is a probability distribution over the positive and negative classes, to which we apply a softmax function. The complete system is illustrated in Figure

4.1. This architecture has been proven to give good results in previous works [17] and in practice in cyberbullying Kaggle competitions [32, 36]. Compared to other state-of-the-art architectures, such as Bidirectional Encoder Representations from Transformers (BERT) or other transformers architectures, it has slightly worse performance but is much faster to train and less demanding in computational resources.

We train our model using the Fused Adam optimiser from Nvidia's Apex library and half-precision [42] to reduce training time. We use a learning rate of 0.05, a decaying factor of 0.6 every epoch and cross-entropy loss. We use a batch size of 128 messages and train on each dataset for 15 epochs.

4.2 Results and Analysis

We present in Tables 4.1 and 4.2 the precision and recall performance of our model when trained using each of our training and validation datasets and tested on each of our datasets. These two metrics are the most important ones to optimize for cyberbullying detection: an ideal cyberbullying filter will block all cyberbullying messages and no messages that are not cyberbullying. Precision is the proportion of messages classified as the positive class that actually belong to it, and recall is the proportion of positive-class messages that are detected as such. While the F1-score combines both metrics, the averaging masks the performance details. More detailed results, including the F1-scores, can be found in Appendix A.

As expected, we find that classifiers trained on each dataset have wildly varying performances on other test datasets, and can see their precision or recall drop by as much as 0.8. However, the results are far from uniformly bad. The top-performing classifier on each test dataset (marked in bold in the tables) is not always the one trained on it, and some classifiers perform as well or even better on other datasets compared to their own test dataset. This means that some generalization of cyberbullying detection is possible. Interestingly, good candidates for generalization are not related to similar data sources: datasets A and B both come from Twitter and datasets F and H both come from Wikipedia talk pages, and while the classifier trained on dataset B does perform better on dataset A than others, the other three show no special affinity for their similar-source datasets. Likewise, modeling a similar trace of cyberbullying does not guarantee generalization: datasets A and E both focus on hate speech, yet they each have very poor recall on the other's test set, meaning they cannot accurately recognize messages the other labels as hate speech.

Looking at Table 4.1, we can see that every model achieves an average precision between 0.45 and 0.63, meaning roughly half the messages each one labels as positive class actually belongs to the negative class. This is a direct consequence of the problem highlighted by the word vectors in Section 2.2 and illustrated in Figure 2.1: a lot of neutral words with similar meanings are observed in multiple datasets but in the positive class of one and the negative

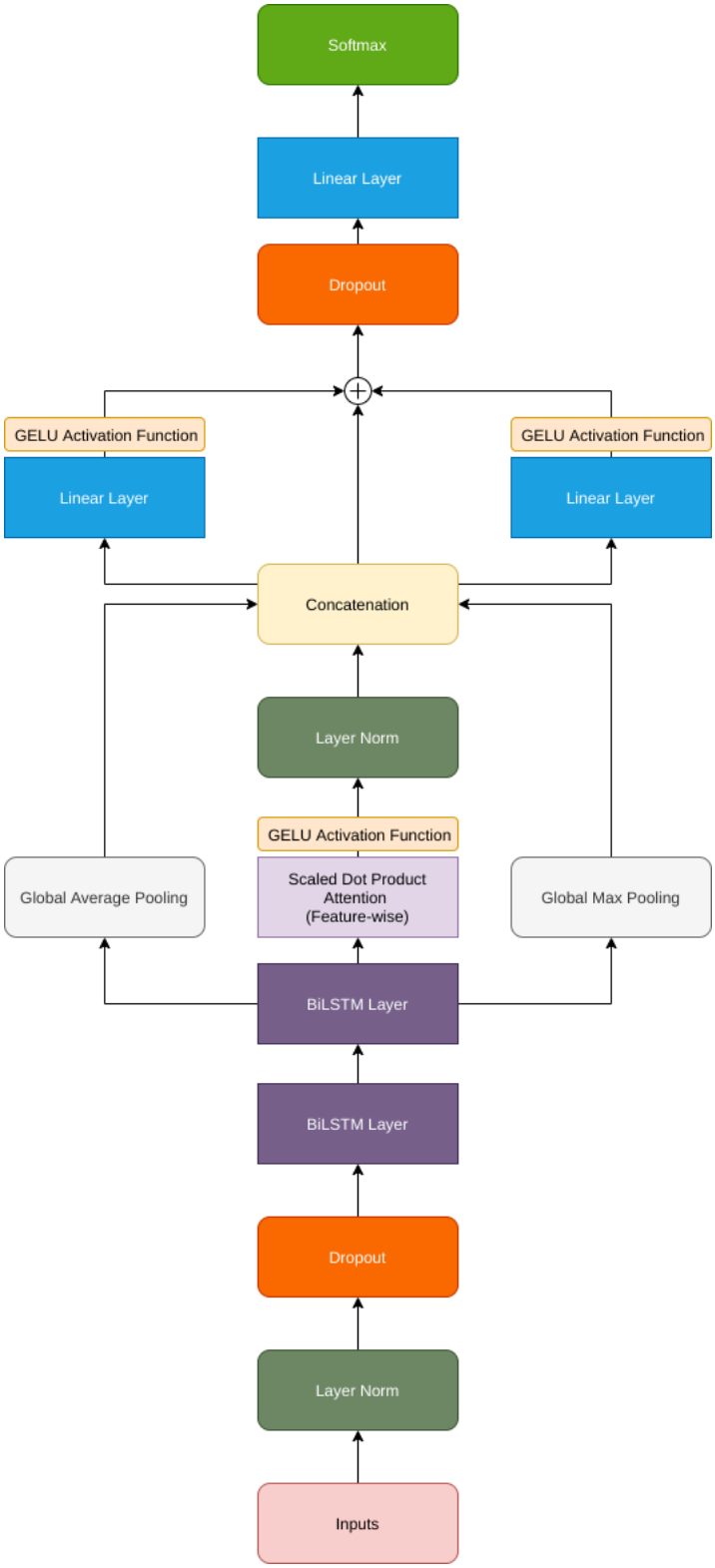

Figure 4.1: Attention BiLSTM Architecture.

class of the other. When such words occur in the training set of one dataset and the test set of another, it causes the test message to be misclassified.

Looking next at recall results, we can see that classifiers often perform worse on other datasets. This is especially true for the classifiers trained on datasets B, C, D, and E; each of them picks out less than half of the positive messages on average across datasets. These are also our four smallest datasets, with less than 15,000 messages each and limited vocabulary in 2.2. The limited variety of messages they have been trained on, compounded by the differences between datasets shown in Section 2.2, means filters trained on these datasets cannot generalize to new situations. By contrast, datasets F and G achieve the highest recall scores, and are able to pick out positive-class messages in other datasets despite their differences. These are also the two largest datasets, and the two that label the largest range of cyberbullying behaviors (six each, compared to one or two in the other datasets). Having seen a greater variety of cyberbullying messages has allowed them to generalize better to new datasets.

It is interesting to note that the classifier trained on dataset C has learned a high-precision and low-recall model. Dataset C was labeled in a more flexible way, by only asking annotators to judge whether they felt a message qualifies as cyberbullying or not, compared to other datasets that gave clear labeling instructions, targeted specific behaviors, or collected messages that used specific bullying keywords or from sources that are known to be toxic. This resulted in a dataset with a very narrow definition of bullying: the intersection where a majority of annotators agree bullying is present contains clear and unambiguous cases (high precision) but more ambiguous cases are missed (low recall). This highlights the benefits of clearly defining the problem behaviors the cyberbullying filter should catch, rather than using a "I know it when I see it" approach.

Looking at performance across test datasets, we find that dataset A is the one on which systems perform the best: all models achieve their highest precision on it, and all but two models achieving better than 0.5 recall. This is because it is the only dataset imbalanced in favour of the positive class. This makes the classification task easier; classifiers can be less discriminating while still avoiding mislabeling negative-class messages in this dataset in a way they cannot in other datasets where positive-class messages are a rare exception. Every classifier achieves a significantly lower precision on every other dataset, including the one it is trained for, indicating that all classifiers routinely mislabel negative-class messages as positive class in all datasets except in dataset A where negative-class messages are just too rare. On the other hand, models achieve most of the lowest precision and recall scores on dataset E. In fact, aside from the model trained specifically on dataset E, every model has trouble with that dataset, with on average only one-third of labeled messages being actually positive-class and one-sixth of positive-class messages being identified as such. This is likely due to that dataset's narrow definition, combined with the nature of the source. Dataset E includes a clear intent to attack in the definition of its positive class, and thus has messages with off-

hand racial slurs are labeled as negative-class due to lack of intent but picked out as positive class by other systems. However, dataset D also requires intent in its positive class, and does not suffer from the same problem. But dataset D was collected from an ordinary social network, while dataset E was collected from Stormfront, a white-supremacist and neo-nazi community. It is not surprising that this dataset includes a lot more negative-class messages with casually aggressive and hateful language that corresponds to positive-class messages of other communities and causes a low precision. The low recall is due to the fact that many of its positive-class messages use racist imagery and idioms, such as attacking people by saying they have "African blood" in them, rather than explicit hate-speech vocabulary.

Table 4.1: Cross-dataset precision

		Test dataset								
		A	B	C	D	E	F	G	H	Average
Train dataset	A	0.9829	0.5699	0.3077	0.5310	0.2742	0.5550	0.4807	0.7686	0.5588
	B	0.9623	**0.7650**	0.2742	0.3480	0.2222	0.4434	0.2335	0.4297	0.4598
	C	**0.9871**	0.6899	**0.5270**	0.6315	**0.6605**	**0.6511**	0.4719	**0.8737**	**0.6302**
	D	0.9741	0.4522	0.1855	**0.7473**	0.1887	0.5868	0.3763	0.7708	0.5352
	E	0.9654	0.7200	0.4130	0.4786	0.5463	0.4846	0.2361	0.6437	0.5610
	F	0.9529	0.5134	0.2127	0.5120	0.3608	0.5134	0.6208	0.7726	0.5573
	G	0.9676	0.5198	0.3012	0.5539	0.3737	0.5775	**0.8019**	0.8055	0.6126
	H	0.9841	0.4294	0.2637	0.5572	0.2292	0.6089	0.6765	0.8387	0.5734
Average		0.9721	0.5845	0.3106	0.5449	0.3570	0.5526	0.4872	0.7379	

Table 4.2: Cross-dataset recall

		Test dataset								
		A	B	C	D	E	F	G	H	Average
Train dataset	A	**0.9749**	0.2119	0.6471	0.6667	0.1717	0.6458	0.1364	0.6251	0.5099
	B	0.5312	**0.7842**	0.2500	0.1140	0.0202	0.1104	0.0360	0.1071	0.2441
	C	0.6298	0.1715	0.5735	0.6825	0.0606	0.4703	0.1169	0.4683	0.3967
	D	0.4360	0.1002	0.6765	0.7042	0.1010	0.4767	0.1996	0.5250	0.4024
	E	0.1484	0.0694	0.2794	0.2424	**0.5960**	0.1861	0.2357	0.1698	0.2409
	F	0.9280	0.2948	**0.8382**	**0.8629**	0.3535	**0.9300**	0.4862	**0.8625**	**0.6945**
	G	0.8091	0.2524	0.7353	0.7864	0.3737	0.8935	**0.5740**	0.7561	0.6476
	H	0.7458	0.1464	0.7794	0.8153	0.1111	0.7935	0.2993	0.7725	0.5579
Average		0.6504	0.2539	0.5974	0.6093	0.2235	0.5633	0.2605	0.5358	

4.3 Performance in Relation to Similarity between Datasets

In Figure 4.2, we consider each of our eight classifiers when tested on each of the eight datasets. The color of each of the 64 points indicates which dataset it was trained on, and its X and Y position indicates the cosine similarity between the pairs of positive and negative class datasets from Table 2.2 respectively. The Z axis measures the AUC of the test, or the

probability of ranking a randomly-chosen positive message higher in positive class than a negative message. Taken together, the figure plots the confidence of the classification given the dataset similarity.

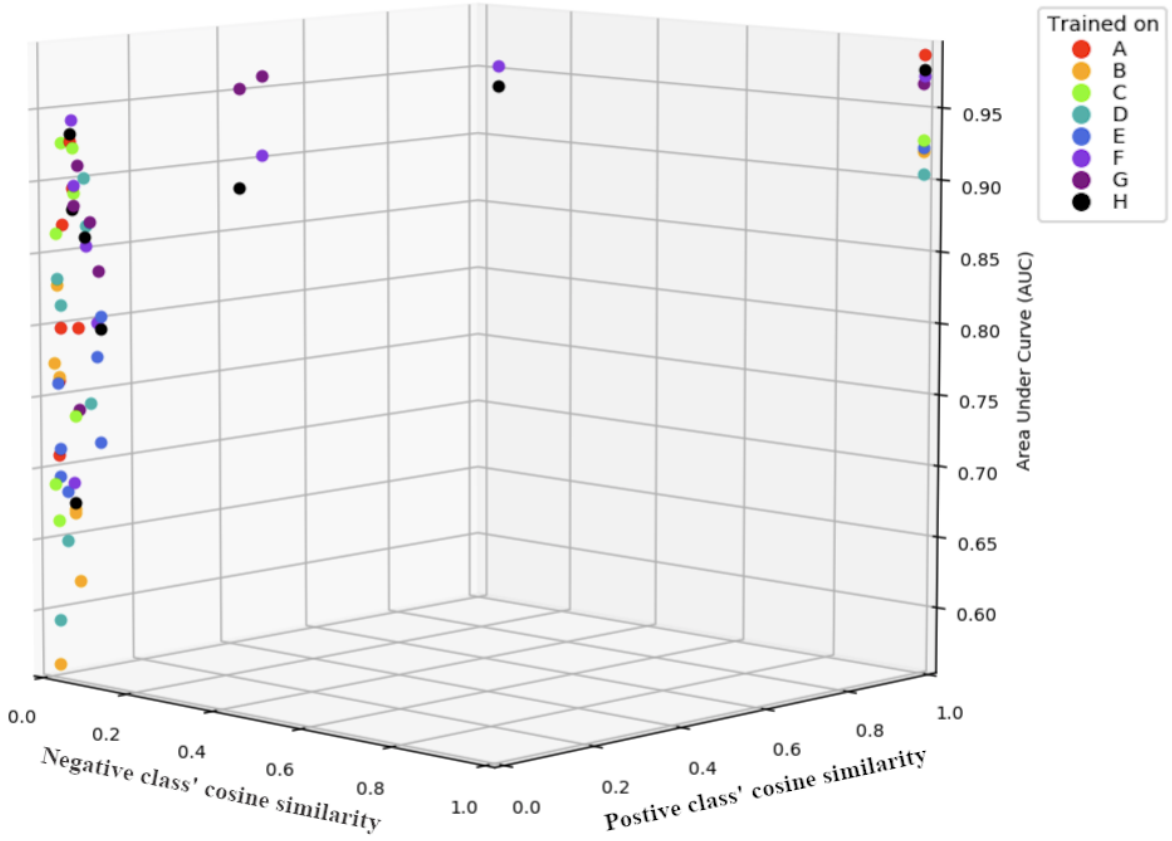

Figure 4.2: AUC of each test given the cosine similarity of the positive and negative datasets.

We can see three clusters of points in Figure 4.2: a tight group near the (0,0) cosine similarity, one at (1,1) similarity, and a scattering of intermediate points. The cluster at similarity (1,1) contains the eight points where a classifier is trained and tested on the same dataset. That cluster has the best AUC score, since every classifier is optimized to its dataset. The scatter of six intermediate points are for classifiers trained on one of datasets F, G and H and tested on each of the other two. As shown in Table 2.2, these datasets are the most similar to each other, and this figure shows that their out-of-domain AUC rivals that of the in-domain (1,1) cluster. Even a limited similarity between the language used in two datasets allows a classifier trained on one to perform well on the other. Finally, all other test results, 50 of our 64 points, are in the (0,0) cluster. These points cover the largest range of performances, from an AUC of 0.55 (almost random chance classification) to 0.95 (surpassing the lower half of the other two clusters). There is no coherence between datasets: the classifiers trained on datasets B and D have the worst AUC performances but also sometimes perform in the low-0.8 range, and classifiers trained on datasets F, G and H get some of the best results in

the 0.90 to 0.95 range but also sometimes perform in the high-0.6 AUC range. Given low similarity between datasets, the performance of a classifier trained on one when applied to the other could be almost anything, and is unpredictable. Given that low similarity is the norm, this further highlights the unreliability of transferring a pre-trained cyberbullying classifier to a new community.

Chapter 5

Ensemble Models Experiments

As the previous chapter has demonstrated, it is difficult to use a cyberbullying detector trained on one corpus to detect problem messages in another. In this chapter, we will experiment instead with strategies to combine a set of detectors, each trained on a different corpus, into an accurate general-domain detector. To explore this question, we will compare different ensemble model architectures built from our individual models.

5.1 Ensemble Models

We implement five ensemble models that combine our existing classifiers in different ways without retraining them.

Linear layer (LL)

In this model, the outputs of the eight individual classifiers (after the softmax function) are combined in a linear layer. This layer is trained using the same hyperparameters as the individual models, and a dropout before the linear layer ensures it does not overfit to a single model's decision. This training step only learns the linear layer's weights.

Democratic voting (DV)

Many of the datasets are labeled independently by multiple annotators, and receive the labels chosen by the majority. We sought to replicate this logic in this ensemble model. Each of our eight classifiers casts a vote based on its classification of a message, and the winning class is simply the one with the most votes. This is akin to having a board of experts each review a message individually, then casting a vote on its class.

Sum voting (SV)

This is a variation of the DV model that takes into account the varying levels of confidence of each model. Instead of an all-or-nothing vote for one class, each classifier votes for both the negative and positive class with the probability it assigns to each class after the softmax function. A classifier that is confident in its result will cast a strong vote for that class while one that is uncertain will cast almost equal votes for both classes and have little influence on the final decision; however, several weak votes in one class may still overrule a single strong vote in the other. In our board of experts metaphor, this is akin to the experts arguing on the class of the message based on the strength of their expertise.

Maximum wins (MW)

This ensemble model picks the classifier with the maximum confidence in its output and assigns the message to its class. In our board of experts metaphor, this is equivalent to having the board examine a message and defer to the expert on that particular message.

Thresholding

In this ensemble model, if any one of the eight classifiers identifies a message as positive class with a confidence above a threshold, it is labeled as such regardless of the output of the other seven classifiers. We implemented two variations of this classifier, one with the confidence threshold at 0.5 (T0.5), which is the lowest possible confidence for a classifier to assign a message to the positive class. The other uses a threshold of 0.95 (T0.95), and if no classifier marks a message as positive with that confidence threshold the ensemble defaults to MW. The first version will thus represent extreme paranoia, where the slightest hint of cyberbullying marks a message as positive class, and the other is a paranoid version of MW.

Dataset merger (DM)

As a baseline, we merged together all eight datasets and trained a new classifier on this dataset, using the same training setup described in chapter 4 and used for the individual classifiers.

5.2 Results and Analysis

Tables 5.1 and 5.2 give the precision and recall value of each ensemble technique when applied to each of our test datasets. More detailed results can be found in Appendix A.

Compared to Tables 4.1 and 4.2, we can see the ensemble models have generally better precision and worse recall. The average F1-scores, given in Table 5.3 for completeness, of the ensemble models are comparable or better to those of the individual classifiers. This

means that combining the information individual classifiers learned from training on different datasets improves the overall performances.

Looking at the performance of individual classifiers, we can see that the three voting classifiers (DV, SV and MW) have similar behaviors: they achieve some of the best precision scores and worst recall scores of all systems. This indicates that most classifiers mislabel most positive-class messages: either those messages have features of the negative class or they lack features of the positive class each classifier is trained for, or both. As a result, when the ensemble decides a message belongs in the positive class, it is usually right. However, most positive-class messages are only recognized by a minority of or by low-confidence classifiers, and thus recall is low.

The LL ensemble also combines the outputs of the eight classifiers, but using a linear layer trained to weight their individual outputs and optimize the decision. It achieves slightly worse precision but much better recall than the DV, SV and MW models. It thus behaves in the opposite manner: it catches a lot more positive-class messages, but mislabels a few more negative-class messages as well. Comparing to the results of Chapter 4, LL actually outperforms all but one of the individual classifiers in precision and in recall and all of them in F1-score, again confirming that there is knowledge to be gained by combining the classifiers. It can be seen that the LL behaves very similarly to the classifier trained on dataset G, achieving just a few points better results in almost every experiment. This is easily explained: the classifier trained on dataset G achieves the best precision and recall on test dataset G, which is the largest dataset, so our linear layer was biased for it. It nonetheless learned to use the outputs of the other classifiers to temper the decision of classifier G and improve its performance in most tests. The most notable exception is actually the recall on test dataset G, where LL performs noticeably worse than classifier G. In that case, classifier G was already very good at picking out positive-class messages, and the other classifiers considered in the linear layer only misclassify the message as negative class.

The paranoid T0.5 model achieves the top recall and lowest precision scores by wide margins. This means that most positive-class messages are labeled as such by at least one classifier, but so are a lot of negative-class messages. If used in practice, this system would create a very clean community mostly devoid of cyberbullying, but would also strongly restrict legitimate conversations. By increasing the decision threshold, T0.95 limits its positive class to messages any one classifier gets a strong signal from. This increases precision in all tests compared to T0.5, as mildly positive messages are no longer marked in the positive class. However, the recall value decreases sharply compared to the T0.5 result, indicating that there are a lot of positive-class messages that not a single classifier can confidently recognize. Interestingly, the performance of T0.95 seems closer to LL than to T0.5.

Interestingly, the DM approach seems to give the best overall performance. It surpasses LL

and T0.95 in precision and is second only to T0.5 in recall, and while it does not match DV, SV and MW in precision it is only 0.04 behind them without suffering from a drop in recall like they do. In terms of F1-score, it is also our best classifier. This indicates that the best way to combine the information from multiple datasets is not by combining multiple individual classifiers but by combining all datasets into a single classifier. This empirical conclusion stems also from our earlier analysis in Chapter 4, where we saw that diversity of language use and of cyberbullying behaviors was key to achieving good results, and similarity of language was important for generalization of the system. By combining all datasets together, the DM classifier is necessarily trained on the largest possible vocabulary and the largest set of different behaviors, and will have vocabulary similarity to all datasets. Moreover, this merger will neutralize the problem exposed in Figure 2.1, of having similar neutral-meaning words that appear only in the vocabulary of the positive class in one dataset and of the negative class in the other and thus confuse the classification. After the merger, these words appear in both classes in the unified dataset and no longer have a strong influence the classification. However, there are two major practical drawbacks to this solution. The first is that a larger dataset means a larger training time; training the DM classifier takes longer than training any of the individual classifiers or building the ensemble models. Secondly, there is no re-usability. If a new dataset is to be integrated into the system, with an ensemble model it is only a matter of training a new individual classifier and incorporating it, while with the DM approach the entire classifier needs to be retrained with the newly-augmented merged dataset. If this is a problem, for instance for a system that routinely needs to adapt to new communities or changes in the existing ones, then the LL or T0.95 ensembles may be preferable alternatives.

Looking at the results per dataset, a lot of our previous observations from Chapter 4 remain true. All systems continue to perform better on dataset A, which is easier to handle due to being imbalanced in favor of the positive class, and all systems struggle on dataset E, indicating just how unusual its breed of hate speech is. In fact the only system to achieve a good recall on that dataset is T0.5, meaning the only way to filter a white-supremacist community seems to be by taking no chances and implementing a zero-tolerance approach.

In table 5.1, the ensemble models with voting techniques, namely sum voting, max voting and democratic voting, have an overall higher precision than the other ensemble models. These techniques show there is knowledge gained when combining the individual models' confidences or votes. However, the performance of the ensemble model with a toxicity threshold at 0.95 achieves a much higher precision than the other ensemble model on the racism and sexism dataset, demonstrating the individual models have specific knowledge that may not be shared when used in a voting manner. There are also noticeable gains between the precision of the ensemble model with a threshold at 0.5 and the variant with a threshold at 0.95, meaning that even though models agree in general, there is seldom a unanimous consensus.

In table 5.2, the three voting ensemble models have noticeably lower recall than the other

techniques. Moreover, the democratic voting is even lower than the sum and max voting. Since DV is the only one not to consider the confidence of each classifier's vote, this seems to indicate that several classifiers are casting a low-confidence vote for a class. This is consistent with the results of the T0.5 model, which clearly shows that there is at least one low-confidence vote for the positive class for many messages.

Table 5.1: Ensemble models precision

	Test dataset								
	A	B	C	D	E	F	G	H	Average
LL	0.9718	0.5627	0.2804	0.5521	0.3766	0.5862	0.8228	0.8221	0.6218
DV	**0.9938**	0.6250	**0.4455**	**0.6803**	0.3043	**0.7786**	0.8734	**0.9337**	**0.7043**
SV	0.9926	0.5649	0.3950	0.6233	0.4138	0.7523	**0.8926**	0.9280	0.6953
MW	0.9906	0.6544	0.3520	0.6029	**0.4583**	0.7146	0.8872	0.9045	0.6956
T0.5	0.9256	0.6145	0.1429	0.4363	0.3047	0.3964	0.3315	0.5756	0.4659
T0.95	0.9759	0.6949	0.3212	0.5654	0.3696	0.6043	0.8202	0.8205	0.6465
DM	0.9759	**0.7169**	0.3520	0.6288	0.3958	0.5844	0.8137	0.8253	0.6616
Average	0.9752	0.6999	0.3270	0.5816	0.3747	0.6310	0.7773	0.8300	

Table 5.2: Ensemble models recall

	Test dataset								
	A	B	C	D	E	F	G	H	Average
LL	0.8656	0.3025	0.7794	0.8182	0.2929	0.9111	0.4977	0.7792	0.6558
DV	0.6931	0.1349	0.6618	0.6941	0.0707	0.5175	0.1124	0.5210	0.4257
SV	0.7825	0.1676	0.6912	0.7403	0.1212	0.6281	0.1624	0.6226	0.4895
MW	0.8705	0.2736	0.6471	0.6681	0.1111	0.5766	0.1631	0.5689	0.4849
T0.5	**0.9928**	**0.8324**	**0.9118**	**0.9293**	**0.7172**	**0.9787**	**0.7539**	**0.9120**	**0.8785**
T0.95	0.9604	0.4740	0.7794	0.7792	0.1717	0.8400	0.3182	0.7496	0.6341
DM	0.9778	0.7418	0.6471	0.7677	0.3838	0.8874	0.5432	0.8204	0.7211
Average	0.8775	0.4181	0.7311	0.7710	0.2669	0.7628	0.3644	0.7105	

To summarize, ensemble classifiers outperform on average the precision, recall and F1-score of individual models, demonstrating there is some gained knowledge to combining and weighting the models. However, the best performance is obtained by training a model on the fusion of all datasets, rather than by combining a set of specialized classifiers.

Table 5.3: Average F1-scores of all classifiers

Training corpus or ensemble method	F1
A	0.5008
B	0.2925
C	0.4611
D	0.4157
E	0.2978
F	0.5845
G	0.6029
H	0.5226
LL	0.6055
DV	0.4828
SV	0.5257
MW	0.5242
T0.5	0.5830
T0.95	0.6027
DM	**0.6778**

Conclusion

The fight against cyberbullying is a challenge of major social importance, and many datasets labeling behaviors associated with cyberbullying are available online to help train filters. In this thesis, we conducted an in-depth study of the relationship between eight of these datasets and the systems that can be trained from them. First, we studied the datasets themselves, and what they tell us about cyberbullying behaviors. Next we studied the similarity in vocabulary between the datasets. Using our own training framework, we then trained deep neural network systems on each dataset and used them to study how they can be transferred from one domain to another. Finally, we studied approaches for combining the classifiers into ensemble models.

The thesis has highlighted four major conclusions. First, there is little agreement on the definition of cyberbullying, the behaviors that comprise it, or how to measure and label them. For instance, hate speech is a recurring trace of cyberbullying in several datasets, but its precise definition varies so much than a classifier trained on one hate-speech dataset can fail to pick it up in another. Our second conclusion is that there is very little language in common between datasets, and what there is is often labeled in contradictory ways, which makes transferring systems from one context to another difficult. In practice, this means a cyberbullying filter built for one community cannot be easily applied to another, and when done the results can be unpredictable. Our third conclusion is that the condition to facilitate transferability is to have a system trained on as diverse a dataset as possible, both in terms of language use and in terms of traces of cyberbullying labeled. This leads into our last conclusion, that if one wishes to combine the knowledge from different datasets in a unified system, the best way of doing this is to merge the datasets and train a single system. This conclusion confirms the claim from the authors of [16] stating that an ensemble model should not be preferred over a model trained on the combined datasets.

However, merging multiple datasets does not solve the underlying problem, the limited vocabulary of the datasets which leads to neutral words being observed exclusively in one class and becoming false classification signals. No additions to the dataset can completely solve that problem. We believe future work should focus on alternative solutions. One possible solution is data augmentation. By replacing words in the messages by nearby words in word-

embedding space, we could create a dataset that more thoroughly explores the vocabulary space. These synonyms would bridge the language gap between neutral words seen only in one class, and would also create a vocabulary buffer around true positive or negative words and strengthen their predictive power. To preserve the meaning of the message in this enhancement step, one could use a context-based word embedding architecture such as the skipgram architecture [43], which will insure context-based synonyms replacement.

Appendix A

Expanded Generalization Results

Table A.1: Cross-dataset Accuracy

		Test dataset								
		A	B	C	D	E	F	G	H	Average
Train dataset	A	**0.9649**	0.6942	0.9038	0.7586	0.8840	0.9149	0.9203	0.9330	**0.8717**
	B	0.5914	**0.8527**	0.9249	0.7121	0.9050	0.8997	0.9147	0.8769	0.8347
	C	0.6841	0.7079	**0.9499**	0.8126	0.8932	0.9247	0.9201	0.9287	0.8527
	D	0.5196	0.6706	0.8247	**0.8602**	0.8795	0.9162	0.9108	0.9249	0.8133
	E	0.2848	0.6911	0.9405	0.7325	**0.9187**	0.9013	0.8796	0.8901	0.7798
	F	0.9016	0.6824	0.8263	0.7488	0.8849	0.9072	0.9361	0.9534	0.8551
	G	0.8181	0.6837	0.8951	0.7782	0.8868	**0.9258**	**0.9552**	0.9493	0.8615
	H	0.7777	0.6619	0.8725	0.7820	0.8858	0.9301	0.9335	**0.9553**	0.8499
Average		0.6928	0.7056	0.8922	0.7731	0.8922	0.9150	0.9213	0.9265	

Table A.2: Ensemble Models Accuracy

		Test dataset								
		A	B	C	D	E	F	G	H	Average
Train dataset	LL	0.8669	0.6992	0.8818	0.7786	0.8922	0.9286	0.9520	0.9537	0.8691
	DV	0.7402	0.6948	**0.9382**	**0.8345**	0.9014	**0.9386**	0.9288	0.9386	0.8644
	SV	0.8136	0.6899	0.9272	0.8149	0.9050	0.9435	0.9324	0.9494	0.8720
	MW	0.8850	0.7191	0.9178	0.7979	**0.9078**	0.9362	0.9324	0.9416	0.8797
	T0.5	0.9274	0.7775	0.7042	0.6672	0.8265	0.8525	0.8608	0.9096	0.8157
	T0.95	0.9472	0.7632	0.9006	0.7854	0.8986	0.9307	0.9408	0.9507	0.8897
	DM	**0.9613**	**0.8222**	0.9178	0.8206	0.8913	0.9274	**0.9542**	**0.9580**	**0.9066**
Average		0.8774	0.7380	0.8840	0.7856	0.8890	0.9225	0.9288	0.9431	

Table A.3: Cross-dataset F1-Score

		Test dataset								
		A	B	C	D	E	F	G	H	Average
Train dataset	A	**0.9789**	0.3090	0.4171	0.5912	0.2112	0.5970	0.2125	0.6895	0.5008
	B	0.6845	**0.7745**	0.2615	0.1717	0.0370	0.1767	0.0624	0.1714	0.2925
	C	0.7690	0.2747	**0.5493**	0.6560	0.0930	0.5494	0.1874	0.6098	0.4611
	D	0.6023	0.1640	0.2911	**0.7251**	0.1316	0.5260	0.2608	0.6246	0.4157
	E	0.2572	0.1265	0.3333	0.3218	**0.5700**	0.2690	0.2359	0.2688	0.2978
	F	0.9403	0.3745	0.3393	0.6427	0.3571	0.6616	0.5454	**0.8151**	0.5845
	G	0.8813	0.3398	0.4274	0.6500	0.3737	**0.7015**	0.**6690**	0.7800	**0.6029**
	H	0.8485	0.2184	0.3941	0.6620	0.1497	0.6891	0.4150	0.8042	0.5226
Average		0.7452	0.3227	0.3766	0.5526	0.2404	0.5213	0.3235	0.5954	

Table A.4: Ensemble Models F1-Score

		Test dataset								
		A	B	C	D	E	F	G	H	Average
Train dataset	LL	0.9156	0.3935	0.4125	0.6593	0.3295	**0.7134**	**0.6203**	0.8001	0.6055
	DV	0.8166	0.2219	**0.5325**	0.6871	0.1148	0.6218	0.1991	0.6688	0.4828
	SV	0.8751	0.2585	0.5027	0.6768	0.1875	0.6846	0.2748	0.7452	0.5257
	MV	0.9267	0.3859	0.4560	0.6338	0.1789	0.6382	0.2755	0.6985	0.5242
	T0.5	0.9580	0.7070	0.2470	0.5938	**0.4277**	0.5643	0.4605	0.7058	0.5830
	T0.95	0.9681	0.5636	0.4549	0.6553	0.2345	0.7029	0.4585	0.7834	0.6027
	DM	**0.9768**	**0.7292**	0.4560	**0.6914**	0.3897	0.7047	0.6515	**0.8228**	**0.6778**
Average		0.9196	0.4656	0.4374	0.6568	0.2661	0.6614	0.4200	0.7464	

Table A.5: Cross-dataset Area Under Curve (AUC)

		Test dataset								
		A	B	C	D	E	F	G	H	Average
Train dataset	A	**0.9870**	0.7097	0.8706	0.7989	0.7619	0.8960	0.7987	0.9279	0.8438
	B	0.8291	**0.9195**	0.7742	0.5626	0.7648	0.6723	0.6214	0.6694	0.7267
	C	0.9269	0.6896	**0.9281**	0.8648	0.6642	0.8931	0.7385	0.9248	0.8288
	D	0.8141	0.5938	0.8334	**0.9040**	0.6502	0.8696	0.7460	0.9034	0.7893
	E	0.7146	0.6946	0.7605	0.6841	**0.9221**	0.7804	0.7192	0.8083	0.7605
	F	0.9427	0.6904	0.8984	0.8557	0.8040	0.9726	0.9211	**0.9798**	0.8831
	G	0.9112	0.7412	0.8846	0.8726	0.8387	**0.9749**	**0.9668**	0.9666	**0.8946**
	H	0.9331	0.6769	0.8817	0.8622	0.7997	0.9656	0.8984	0.9767	0.8743
Average		0.8823	0.7144	0.8540	0.8006	0.7757	0.8781	0.8013	0.8946	

Table A.6: Ensemble Models Area Under Curve (AUC)

		Test dataset								
		A	B	C	D	E	F	G	H	Average
Train dataset	LL	0.9599	0.8242	0.9138	0.8614	0.8582	**0.9767**	0.9460	0.9739	0.9143
	DV	0.9643	0.7873	0.9013	0.8642	0.7900	0.9612	0.8397	0.9420	0.8812
	SV	0.9721	0.8181	**0.9189**	0.8799	**0.8634**	0.9724	0.9273	0.9745	0.9158
	MV	0.9694	0.8427	0.9161	0.8436	0.8542	0.9661	0.9041	0.9670	0.9079
	T0.5	0.9741	0.8390	0.9133	0.8500	0.8529	0.9706	0.9037	0.9685	0.9090
	T0.95	0.9742	0.8391	0.9143	0.8499	0.8535	0.9706	0.9042	0.9686	0.9093
	DM	**0.9820**	**0.8935**	0.9057	**0.8899**	0.8536	0.9747	**0.9654**	**0.9788**	**0.9304**
Average		0.9709	0.8348	0.9119	0.8627	0.8465	0.9703	0.9129	0.9676	

Appendix B

Top 30 words of each dataset

Table B.1: Top 30 words of each dataset according to the TFIDF per Dataset's Class

Class	A	B	C	D
Neg.	'iubb', 'cantucimblonde', 'kacado', 'farewellcaptain', 'revkahjc', 'rejectedpeanutsspecials', 'busboysandpoets', 'itsfoodporn', 'blessjesus', 'virginiacity', 'travelnevada', 'mcbob', 'kejriwal', 'shiner', 'wcvb', 'pacers', 'puremonotheist', 'uniteblue', 'ariennajanee', 'thehermancain', 'springtraining', 'brandontierney', 'billythathird', 'milca', 'pacernation', 'victoriakuhlman', 'wizkid', 'daggerbyte', 'brianndominguez', 'tyzebruh'	'freebsdgirl', 'spacekatgal', 'thequinnspiracy', 'chriswarcraft', 'girlziplocked', 'ggautoblocker', 'peerworker', 'srhbutts', 'sschinke', 'rjennromao', 'ameliagreenhall', 'novorossiyan', 'gbabeuf', 'gbazov', 'newscoverup', 'hypatiadotca', 'dylanw', 'evvykube', 'athenahollow', 'ypj', 'ashleylynch', 'kaitlynburnell', 'shaz', 'colonelkickhead', 'wadhwa', 'chuckpfarrer', 'glennf', 'sarahjeong', 'fourinhand', 'metroidthief'	'muj', 'trebla', 'wasup', 'sexxii', 'hbu', 'suree', 'awsum', 'naomily', 'dontt', 'yhu', 'tabiisaninja', 'yupz', 'yhur', 'watev', 'nopeee', 'ermmm', 'ididnt', 'emz', 'snog', 'lauraa', 'hadirty', 'spamledge', 'chuz', 'lolr', 'kidn', 'loveee', 'awhh', 'ebuddy', 'funi', 'kute'	'mrzlgv', 'nhttp', 'ufeff', 'xbf', 'nhow', 'yhelahaq', 'nobjustryan', 'yinzerinct', 'nread', 'npeople', 'qkujt', 'yxvuisfr', 'srguvysy', 'qzlij', 'nonstopdrivel', 'juego', 'xea', 'xean', 'xecnh', 'nwho', 'nbesides', 'nfrom', 'nshe', 'nbtw', 'nlike', 'qzcvl', 'ufeffhttp', 'unre', 'immdufcc', 'cellulosic'
Pos.	'niggahs', 'joebudden', 'wyattnuckels', 'causewereguys', 'vianawf', 'kinghorsedick', 'oomf', 'xdsmooth', 'grizzboadams', 'ctfu', 'voiceofdstreetz', 'iont', 'jawshoeeahhh', 'sbsylvester', 'caymarieee', 'cuhcuhcuh', 'mrmooncricket', 'boosie', 'darkskin', 'thecoreyholcomb', 'bouta', 'lebronvuitton', 'mikediggem', 'taxstone', 'lilduval', 'cuffin', 'oskzilla', 'tryn', 'bbluedreamm', 'morbidmermaid'	'womenagainstfeminism', 'questionsformen', 'shermertron', 'humanistfury', 'bristolben', 'feministlah', 'greenweiner', 'boxedariel', 'ajwatamr', 'adviceforyoungfeminists', 'chubssays', 'anniekfox', 'uberfeminist', 'dqtwitchstream', 'tehmenz', 'ajzionts', 'tamedinsanity', 'anonmnom', 'sorryitsaboy', 'amberhasalamb', 'dezzantibus', 'ilivundrurbed', 'oneiorosgrip', 'avacadosoup', 'meninisttweet', 'theinfoislam', 'mgtowknight', 'synthovine', 'juliandavis', 'superjutah'	'haterr', 'tryy', 'fuckinn', 'lifee', 'wrry', 'woood[1], 'kume', 'trickk', 'catchh', 'gigglez', 'callingg', 'yuurr', 'dntt', 'celetics', 'bhee', 'gayy', 'anonymousness', 'betch', 'sayinnnnn', 'blahhhh', 'yaaaahhh', 'thiiss', 'whooss', 'tevon', 'cleaa', 'ugliii', 'worryinn', 'kaydoll', 'bettaa', 'rhubinathy'	'xaf', 'nfact', 'fuuk', 'nseriously', 'murdouchebag', 'teabilly', 'slutchild', 'nbest', 'fkin', 'hannituru', 'nexcept', 'kkkonservative', 'berthole', 'nthings', 'vidalia', 'vadalia', 'klaners', 'yoursilf', 'arealconservati', 'ntill', 'christard', 'nknicks', 'beeyatch', 'smugs', 'nmoron', 'weaman', 'nred', 'nypocrtical', 'bteer', 'nhand'

[1] Letter 'd' repeated 26 time

Table B.2: (Continued) Top 30 words of each dataset

Class	E	F	G	H
Neg.	'hvorostovsky', 'rüütel', 'lambent', 'abqtrib', 'derren', 'sørensen', 'sportsmansguide', 'rlm', 'irishcentral', 'kalispell', 'ð', 'sqwincher', 'halliujah', 'jjt', 'intj', 'extraversion', 'wntube', 'medlem', 'stomper', 'csabi', 'mjodr', 'gliwice', 'nordid', 'ngc', 'mediafire', 'legionism', 'eistlands', 'kuzmin', 'denson', 'flyering'	'deneid', 'saget', 'cheesei', 'failepic', 'gabsadds', 'philippineslong', 'cohanim', 'ekman', 'pagedelete', 'talkstalk', 'jadoon', 'zzz[1], 'cellpadding', 'orderinchaos', 'suro', 'onka', 'dhudhi', 'mainpagebg', 'coibot', 'wilayat', 'townsville', 'khuzaima', 'qutbuddin', 'shukumine', 'gusuku', 'gayvn', 'copyvio', 'gusle', 'chillum', 'tkd'	'A', 'AND' , 'rrsps' , 'THE' ,'HOME' ,'UP', 'AT', 'YOU', 'BY', 'maglev', 'whiteaker', 'dhhl', 'fluorosis', 'guideway', 'rrif', 'hannemann', 'kahala', 'tfsas', 'reit', 'gics', 'galera', 'corvallis', 'FOR', 'CHECK', 'aidea', 'COMPUTER', 'THIS', 'biki', 'MONTH', 'WORKING'	'spamrofl', 'boymamas', 'deletedthis', 'hahaha[2], 'ewfh', 'pagedelete', 'philippineslong', 'cellpadding', 'ayyavazhi', 'cellspacing', 'signalhead', 'broyard', 'tajin', 'pengei', 'crantius', 'colto', 'nocciolini', 'canzo', 'mainpagebg', 'emule', 'copyvio', 'rexcurry', 'nacer', 'gaeltacht', 'în', 'eprocure', 'biggov', 'rauber', 'khagan', 'colspan'
Pos.	'ghlight', 'tiolets', 'westfälische', 'probz', 'dieversity', 'crimusz', 'christentum', 'schlimm', 'moblie', 'crippin', 'mudraces', 'jewlords', 'apelantic', 'rohawa', 'menting', 'jewv', 'mildura', 'proffesionalism', 'intimidae', 'wariors', 'gdyjju', 'proncipal', 'lentin', 'zpridez', 'accommonadate', 'prreachers', 'evoluated', 'unfare', 'serbi', 'bulgayri'	'offfuck', 'sexsex', 'criminalwar', 'securityfuck', 'aidsaids', 'bitchmattythewhite', 'haahhahahah', 'biznitch', 'cuntliz', 'pennnis', 'pneis', 'pensnsnniensnsn', 'itsuck', 'babywhat', 'cuntfranks', 'fuckingabf', 'bongwarriorcongratualtions', 'antivman', 'jforget', 'vuvuzelas', 'fuckbags', 'mothjer', 'youcaltlas', 'fggt', 'shoit', 'yourselfgo', 'marcolfuck', 'bitchmother', 'cuntbag', 'youbollocks'	'israheil', 'nigers', 'duben', 'ofass', 'ohboofreakinghoohillarylost', 'donutholes', 'durrrr', 'bwoop', 'stfd', 'stupiditing', 'praveen', 'becos', 'cafod', 'balki', 'greazy', 'straigtup', 'bullllshit', 'umptions', 'yvoneravone', 'surroagte', 'baleeve', 'bandoliers', 'narcicist', 'koss', 'zinga', 'sihon', 'dnecks', 'holsteins', 'grk', 'rimi'	'fucksex', 'gayi', 'dcks', 'nigganigga', 'yourselfgo', 'marcolfuck', 'youbollocks', 'ullmann', 'shitfuck', 'fggt', 'notrhbysouthbanof', 'hahaha[3], 'centralistst upid', 'mothjer', 'bleachanhero', 'muahahahaha[4], 'cuntbag', 'chocobos', 'gayfrozen', 'bestfrozen', 'phuq', 'caltlas', 'fucky', 'dickface', 'billj', 'retardedyour', 'jpgsuck', 'kente', 'nihgaa', 'suckersyou'

[1] Character 'z' repeated 150 time

[2] Letters 'ha' repeated 26 time

[3, 4] Letters 'ha' repeated 16 time

[*] **Note**: Some of the words only appear once or have very low occurrence in the datasets

Bibliography

[1] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," *CoRR*, vol. abs/1810.04805, 2018. [Online]. Available: http://arxiv.org/abs/1810.04805

[2] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "Distilbert, a distilled version of BERT: smaller, faster, cheaper and lighter," *CoRR*, vol. abs/1910.01108, 2019. [Online]. Available: http://arxiv.org/abs/1910.01108

[3] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, no. 56, pp. 1929–1958, 2014. [Online]. Available: http://jmlr.org/papers/v15/srivastava14a.html

[4] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," *Transactions of the Association for Computational Linguistics*, vol. 5, pp. 135–146, 2017.

[5] J. Pennington, R. Socher, and C. Manning, "GloVe: Global vectors for word representation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics, Oct. 2014, pp. 1532–1543. [Online]. Available: https://www.aclweb.org/anthology/D14-1162

[6] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. Curran Associates, Inc., 2017, pp. 5998–6008. [Online]. Available: https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf

[7] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," 2013.

[8] D. W. Hango, *Cyberbullying and cyberstalking among Internet users aged 15 to 29 in Canada*. Statistics Canada Ottawa, Ontario, 2016.

[9] "Free to play? hate, harassment, and positive social experiences in online games," ADL, Tech. Rep., 2019. [Online]. Available: https://www.adl.org/free-to-play

[10] E. Wulczyn, N. Thain, and L. Dixon, "Ex machina: Personal attacks seen at scale," in *Proceedings of the 26th International Conference on World Wide Web*, ser. WWW '17. International World Wide Web Conferences Steering Committee, 2017, pp. 1391–1399. [Online]. Available: http://doi.org/10.1145/3038912.3052591

[11] P. K. Smith, J. Mahdavi, M. Carvalho, S. Fisher, S. Russell, and N. Tippett, "Cyberbullying: Its nature and impact in secondary school pupils," *Journal of child psychology and psychiatry*, vol. 49, no. 4, pp. 376–385, 2008.

[12] C. V. Hee, E. Lefever, B. Verhoeven, J. Mennes, B. Desmet, G. D. Pauw, W. Daelemans, and V. Hoste, "Detection and fine-grained classification of cyberbullying events," p. 9, 2015.

[13] D. Chatzakou, N. Kourtellis, J. Blackburn, E. De Cristofaro, G. Stringhini, and A. Vakali, "Mean birds: Detecting aggression and bullying on twitter," *arXiv:1702.06877 [cs]*, 2017. [Online]. Available: http://arxiv.org/abs/1702.06877

[14] F. K. Ventirozos, I. Varlamis, and G. Tsatsaronis, "Detecting aggressive behavior in discussion threads using text mining," in *Computational Linguistics and Intelligent Text Processing*, ser. Lecture Notes in Computer Science, A. Gelbukh, Ed. Springer International Publishing, 2018, pp. 420–431.

[15] T. Bin Abdur Rakib and L.-K. Soon, "Using the reddit corpus for cyberbully detection," in *Intelligent Information and Database Systems*, ser. Lecture Notes in Computer Science, N. T. Nguyen, D. H. Hoang, T.-P. Hong, H. Pham, and B. Trawiński, Eds. Springer International Publishing, 2018, pp. 180–189.

[16] C. Emmery, B. Verhoeven, G. D. Pauw, G. Jacobs, C. V. Hee, E. Lefever, B. Desmet, V. Hoste, and W. Daelemans, "Current limitations in cyberbullying detection: on evaluation criteria, reproducibility, and data scarcity," *CoRR*, vol. abs/1910.11922, 2019. [Online]. Available: http://arxiv.org/abs/1910.11922

[17] S. Agrawal and A. Awekar, "Deep learning for detecting cyberbullying across multiple social media platforms," in *Advances in Information Retrieval*, G. Pasi, B. Piwowarski, L. Azzopardi, and A. Hanbury, Eds. Cham: Springer International Publishing, 2018, pp. 141–153.

[18] N. Potha and M. Maragoudakis, "Time series forecasting in cyberbullying data," in *Engineering Applications of Neural Networks*, ser. Communications in Computer and Information Science, L. Iliadis and C. Jayne, Eds. Springer International Publishing, 2015, pp. 289–303.

[19] H. Hosseinmardi, S. A. Mattson, R. I. Rafiq, R. Han, Q. Lv, and S. Mishr, "Prediction of cyberbullying incidents on the instagram social network," *arXiv:1508.06257 [cs]*, 2015. [Online]. Available: http://arxiv.org/abs/1508.06257

[20] Y. N. Silva, D. L. Hall, and C. Rich, "BullyBlocker: toward an interdisciplinary approach to identify cyberbullying," *Social Network Analysis and Mining*, vol. 8, no. 1, p. 18, 2018. [Online]. Available: https://doi.org/10.1007/s13278-018-0496-z

[21] H. Zhong, D. J. Miller, and A. Squicciarini, "Flexible inference for cyberbully incident detection," in *Machine Learning and Knowledge Discovery in Databases*, ser. Lecture Notes in Computer Science, U. Brefeld, E. Curry, E. Daly, B. MacNamee, A. Marascu, F. Pinelli, M. Berlingerio, and N. Hurley, Eds. Springer International Publishing, 2019, pp. 356–371.

[22] J. J. da Silveira Marciano, E. M. A. M. Mendes, and M. F. S. Barroso, "Cyberbullying classification using extreme learning machine applied to portuguese language," in *Computational Neuroscience*, ser. Communications in Computer and Information Science, D. A. C. Barone, E. O. Teles, and C. P. Brackmann, Eds. Springer International Publishing, 2017, pp. 109–117.

[23] M. Ptaszynski, F. Masui, Y. Kimura, R. Rzepka, and K. Araki, "Automatic extraction of harmful sentence patterns with application in cyberbullying detection," in *Human Language Technology. Challenges for Computer Science and Linguistics*, Z. Vetulani, J. Mariani, and M. Kubis, Eds. Cham: Springer International Publishing, 2018, pp. 349–362.

[24] H. Dani, J. Li, and H. Liu, "Sentiment informed cyberbullying detection in social media," in *Machine Learning and Knowledge Discovery in Databases*, ser. Lecture Notes in Computer Science, M. Ceci, J. Hollmén, L. Todorovski, C. Vens, and S. Džeroski, Eds. Springer International Publishing, 2017, pp. 52–67.

[25] E. Raisi and B. Huang, "Cyberbullying identification using participant-vocabulary consistency," *arXiv:1606.08084 [cs, stat]*, 2016. [Online]. Available: http://arxiv.org/abs/1606.08084

[26] C. Van Hee, G. Jacobs, C. Emmery, B. Desmet, E. Lefever, B. Verhoeven, G. De Pauw, W. Daelemans, and V. Hoste, "Automatic detection of cyberbullying in social media text," *PloS one*, vol. 13, no. 10, 2018.

[27] K. Reynolds, A. Kontostathis, and L. Edwards, "Using machine learning to detect cyberbullying," in *2011 10th International Conference on Machine Learning and Applications and Workshops*, vol. 2, 2011, pp. 241–244.

[28] 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*. San Diego, California: Association for Computational Linguistics, June 2016, pp. 88–93. [Online]. Available: http://www.aclweb.org/anthology/N16-2013

[29] J. Bayzick, A. Kontostathis, and L. Edwards, "Detecting the presence of cyberbullying using computer software," 2011.

[30] U. Bretschneider, T. Wöhner, and R. Peters, "Detecting online harassment in social networks," 2014.

[31] J.-M. Xu, K.-S. Jun, X. Zhu, and A. Bellmore, "Learning from bullying traces in social media," in *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Montréal, Canada: Association for Computational Linguistics, Jun. 2012, pp. 656–666. [Online]. Available: https://www.aclweb.org/anthology/N12-1084

[32] Jigsaw, "Toxic Comment Classification Challenge," 2017. [Online]. Available: https://kaggle.com/c/jigsaw-toxic-comment-classification-challenge

[33] T. Davidson, D. Warmsley, M. Macy, and I. Weber, "Automated hate speech detection and the problem of offensive language," in *Proceedings of the 11th International AAAI Conference on Web and Social Media*, ser. ICWSM '17, 2017, pp. 512–515.

[34] Imperium, "Detecting Insults in Social Commentary," 2012. [Online]. Available: https://kaggle.com/c/detecting-insults-in-social-commentary

[35] O. de Gibert, N. Perez, A. García-Pablos, and M. Cuadros, "Hate Speech Dataset from a White Supremacy Forum," in *Proceedings of the 2nd Workshop on Abusive Language Online (ALW2)*. Brussels, Belgium: Association for Computational Linguistics, Oct. 2018, pp. 11–20. [Online]. Available: https://www.aclweb.org/anthology/W18-5102

[36] Jigsaw, "Jigsaw Unintended Bias in Toxicity Classification," 2019. [Online]. Available: https://kaggle.com/c/jigsaw-unintended-bias-in-toxicity-classification

[37] L. van der Maaten and G. Hinton, "Viualizing data using t-sne," *Journal of Machine Learning Research*, vol. 9, pp. 2579–2605, 11 2008.

[38] E. Gamma, R. Helm, R. Johnson, and J. M. Vlissides, *Design Patterns: Elements of Reusable Object-Oriented Software*, 1st ed. Addison-Wesley Professional, 1994. [Online]. Avail-

able: http://www.amazon.com/Design-Patterns-Elements-Reusable-Object-Oriented/dp/0201633612/ref=ntt_at_ep_dpi_1

[39] E. Brassard-Gourdeau and R. Khoury, "Subversive toxicity detection using sentiment information," in *Proceedings of the Third Workshop on Abusive Language Online*, 2019, pp. 1–10.

[40] S. Narang, G. Diamos, E. Elsen, P. Micikevicius, J. Alben, D. Garcia, B. Ginsburg, M. Houston, O. Kuchaiev, G. Venkatesh, and H. Wu, "MIXED PRECISION TRAINING," p. 12, 2018.

[41] D. Hendrycks and K. Gimpel, "Gaussian error linear units (GELUs)," *arXiv:1606.08415 [cs]*, 2018. [Online]. Available: http://arxiv.org/abs/1606.08415

[42] P. Micikevicius, S. Narang, J. Alben, G. Diamos, E. Elsen, D. Garcia, B. Ginsburg, M. Houston, O. Kuchaiev, G. Venkatesh *et al.*, "Mixed precision training," *arXiv preprint arXiv:1710.03740*, 2017.

[43] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in Neural Information Processing Systems 26*, C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2013, pp. 3111–3119. [Online]. Available: http://papers.nips.cc/paper/5021-distributed-representations-of-words-and-phrases-and-their-compositionality.pdf