HATE SPEECH DETECTION USING NATURAL LANGUAGE PROCESSING

A MINI PROJECT REPORT

18CSC304J- COMPILER DESIGN

Submitted by

OM TIWARI(RA2011026010341)
SIDDHARTH S (RA2011026010368)
GEETHA SHASHANK PERICHERLA(RA2011026010380)

Under the guidance of

Dr. Maheshwari A

Assistant Professor, Department of Computer Science and Engineering

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE & ENGINEERING

of

FACULTY OF ENGINEERING AND TECHNOLOGY

S.R.M. Nagar, Kattankulathur, Chengalpattu District

MAY 2022

SRM INSTITUTE OF SCIENCE AND TECHNOLOGY

(Under Section 3 of UGC Act, 1956)

BONAFIDE CERTIFICATE

Certified that Mini project report titled **"HATE SPEECH DETECTION USING NATURAL LANGUAGE PROCESSING "** is the bona fide work of **OM TIWARI (RA2011026010341), SIDDHARTH S (RA2011026010368), GEETHA SHASHANK PERICHERLA (RA2011026010380)** who carried out the minor project under my supervision. Certified further, that to the best of my knowledge, the work reported herein does not form any other project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE

Dr.Maheshwari A
Assistant Professor
Department of Computing Technologies

ABSTRACT

We explore the idea of creating a classifier that can be used to detect presence of hate speech in web discourses such as web forums and blogs. In this work, hate speech problem is abstracted into three main thematic areas of race, nationality and religion. The goal of our research is to create a model classifier that uses sentiment analysis techniques and in particular subjectivity detection to not only detect that a given sentence is subjective but also to identify and rate the polarity of sentiment expressions. We begin by whittling down the document size by removing objective sentences. Then, using subjectivity and semantic features related to hate speech, we create a lexicon that is employed to build a classifier for hate speech detection. Experiments with a hate corpus show significant practical application for a real-world web discourse.Hate speech is one type of harmful online content which directly attacks or promotes hate towards a group or an individual member based on their actual or perceived aspects of identity, such as ethnicity, religion, and sexual orientation. With online hate speech on the rise, its automatic detection as a natural language processing task is gaining increasing interest. However, it is only recently that it has been shown that existing models generalize poorly to unseen data. This survey paper attempts to summarize how generalizable existing hate speech detection models are and the reasons why hate speech models struggle to generalize, sums up existing attempts at addressing the main obstacles, and then proposes directions of future research to improve generalization in hate speech detection. As online content continues to grow, so does the spread of hate speech. We identify and examine challenges faced by online automatic approaches for hate speech detection in text. Among these difficulties are subtleties in language, differing definitions on what constitutes hate speech, and limitations of data availability for training and testing of these systems. Furthermore, many recent approaches suffer from an interpretability problem—that is, it can be difficult to understand why the systems make the decisions that they do. We propose a multi-view SVM approach that achieves near state-of-the-art performance, while being simpler and producing more easily interpretable decisions than neural methods. We also discuss both technical and practical challenges that remain for this task

While favoring communications and easing information sharing, Social Network Sites are also used to launch harmful campaigns against specific groups and individuals. Cyber bullism, incitement to self-harm practices, sexual predation are just some of the severe effects of massive online offensives. Moreover, attacks can be carried out against groups of victims and can degenerate into physical violence. In this work, we aim at containing and preventing the alarming diffusion of such hate campaigns. Using Facebook as a benchmark, we consider the textual content of comments appearing on a set of public Italian pages. We first propose a variety of hate categories to distinguish the kind of hate. Crawled comments are then annotated by up to five distinct human annotators, according to the defined taxonomy. Leveraging morpho-syntactic features, sentiment polarity and word embedding lexicons, we design and implement two classifiers for the Italian language, based on different learning algorithms: the first based on Support Vector Machines (SVM) and the second on a particular Recurrent Neural Network named Long Short Term Memory (LSTM).

TABLE OF CONTENTS

LIST OF FIGURES

ABBREVIATIONS

LSTM Long short term memory

GRU Gated Recurrent Unit

AI Artificial Intelligence

BERT Bidirectional Encoder Representations

TRAC Trolling, Aggression and cyberbullying

SVM Support vector machine

CHAPTER 1

INTRODUCTION

The Internet saw a growing body of user-generated content as social media platforms flourished (Schmidt & Wiegand, 2017; Chung et al., 2019). While social media provides a platform for all users to freely express themselves, offensive and harmful contents are not rare and can severely impact user experience and even the civility of a community (Nobata et al., 2016). One type of such harmful content is hate speech, which is speech that directly attacks or promotes hate towards a group or an individual member based on their actual or perceived aspects of identity, such as ethnicity, religion, and sexual orientation (Waseem & Hovy, 2016; Davidson et al., 2017; Founta et al., 2018; Sharma, Agrawal & Shrivastava, 2018). Major social media companies are aware of the harmful nature of hate speech and have policies regarding the moderation of such posts. However, the most commonly used mechanisms are very limited. For example, keyword filters can deal with profanity, but not the nuance in the expression of hate (Gao, Kuppersmith & Huang, 2017). Crowd-sourcing methods (e.g., human moderators, user reporting), on the other hand, do not scale up. This means that by the time that a hateful post gets detected and taken down, it has already made negative impacts (Chen, McKeever & Delany, 2019).

The automatic detection of hate speech is thus an urgent and important task. Since the automatic detection of hate speech was formulated as a task in the early 2010s (Warner & Hirschberg, 2012), the field has been constantly growing along the perceived importance of the task.

Hate crimes are unfortunately nothing new in society. However, social media and other means of online communication have begun playing a larger role in hate crimes. For instance, suspects in several recent hate-related terror attacks had an extensive social media history of hate-related posts, suggesting that social media contributes to their radicalization [1, 2]. In some cases, social media can play an even more direct role; video

footage from the suspect of the 2019 terror attack in Christchurch, New Zealand, was broadcast live on Facebook [2].

Vast online communication forums, including social media, enable users to express themselves freely, at times, anonymously. While the ability to freely express oneself is a human right that should be cherished, inducing and spreading hate towards another group is an abuse of this liberty. For instance, The American Bar Association asserts that in the United States, hate speech is legal and protected by the First Amendment, although not if it directly calls for violence [3]. As such, many online forums such as Facebook, YouTube, and Twitter consider hate speech harmful, and have policies to remove hate speech content [4–6]. Due to societal concern and how widespread hate speech is becoming on the Internet [7], there is strong motivation to study automatic detection of hate speech. By automating its detection, the spread of hateful content can be reduced.

Detecting hate speech is a challenging task, however. First, there are disagreements on how hate speech should be defined. This means that some content can be considered hate speech to some and not to others, based on their respective definitions. We start by covering competing definitions, focusing on the different aspects that contribute to hate speech. We are by no means, nor can we be, comprehensive as new definitions appear regularly. Our aim is simply to illustrate variances highlighting difficulties that arise from such.

Competing definitions provide challenges for evaluation of hate speech detection systems; existing datasets differ in their definition of hate speech, leading to datastets that are not only from different sources, but also capture different information. This can make it difficult to directly access which aspects of hate speech to identify. We discuss the various datasets available to train and measure the performance of hate speech detection systems in the next section. Nuance and subtleties in language provide further challenges in automatic hate speech identification, again depending on the definition.

CHAPTER 2

LITERATURE SURVEY

Although different types of abusive and offensive language are closely related, there are important distinctions to note. Offensive language and abusive language are both used as umbrella terms for harmful content in the context of automatic detection studies. However, while "strongly impolite, rude" and possible use of profanity are seen in the definitions of both (Fortuna & Nunes, 2018), abusive language has a strong component of intentionality (Caselli et al., 2020). Thus, offensive language has a broader scope, and hate speech falls in both categories.

Because of its definition mentioned above, hate speech is also different from other subtypes of offensive language. For example, personal attacks (Wulczyn, Thain & Dixon, 2017) are characterized by being directed at an individual, which is not necessarily motivated by the target's identity. Hate speech is also different from cyberbullying (Zhao, Zhou & Mao, 2016), which is carried out repeatedly and over time against vulnerable victims that cannot defend themselves.1 This paper focuses on hate speech and hate speech datasets, although studies that cover both hate speech and other offensive language are also mentioned.

The ultimate purpose of studying automatic hate speech detection is to facilitate the alleviation of the harms brought by online hate speech. To fulfill this purpose, hate speech detection models need to be able to deal with the constant growth and evolution of hate speech, regardless of its form, target, and speaker.

Recent research has raised concerns on the generalisability of existing models (Swamy, Jamatia & Gambäck, 2019). Despite their impressive performance on their respective test sets, the performance significantly dropped when the models are applied to a different hate speech dataset. This means that the assumption that test data of existing datasets represent the distribution of future cases is not true, and that the generalization performance of existing models have been severely overestimated (Arango, Prez & Poblete, 2020). This lack of generalisability undermines the practical value of these hate speech detection models.

So far, existing research has mainly focused on demonstrating the lack of generalisability (Gröndahl et al., 2018; Swamy, Jamatia & Gambäck, 2019; Wiegand, Ruppenhofer &

Kleinbauer, 2019; Fortuna, Soler-Company & Wanner, 2021), apart from a handful of studies that made individual attempts at addressing aspects of it (Karan & Šnajder, 2018; Waseem, Thorne & Bingel, 2018; Arango, Prez & Poblete, 2020). Recent survey papers on hate speech and abusive language detection (Schmidt & Wiegand, 2017; Fortuna & Nunes, 2018; Al-Hassan & Al-Dossari, 2019; Mishra, Yannakoudakis & Shutova, 2019; Vidgen et al., 2019; Poletto et al., 2020; Vidgen & Derczynski, 2020) have focused on the general trends in this field, mainly by comparing features, algorithms and datasets. Among these, Fortuna & Nunes (2018) provided an in-depth review of definitions, Vidgen et al. (2019) concisely summarized various challenges for the detection of abusive language in general, Poletto et al. (2020) and Vidgen & Derczynski (2020) created extensive lists of resources and benchmark corpora while Al-Hassan & Al-Dossari (2019) focused on the special case of the Arabic language.

This survey paper thus contributes to the literature by providing (1) a comparative summary of existing research that demonstrated the lack of generalisability in hate speech detection models, (2) a systematic analysis of the main obstacles to generalisable hate speech detection and existing attempts to address them, and (3) suggestions for future research to address these obstacles.

This paper is most relevant to any researcher building datasets of, or models to detect, online hate speech, but can also be of use for those who work on other types of abusive or offensive language.

CHAPTER 3

SYSTEM ARCHITECTURE AND DESIGN

i.MODELS

First of all, model performance had been severely over-estimated. This includes existing "state-of-the-art" models and common baselines. Models used in the experiments ranged from neural networks—deep or shallow—to classical machine learning methods, including mixtures of both. When applied cross-dataset, all show a significant performance drop. Performance on a different dataset highlights that the test set of the same dataset does not realistically represent the distribution of unseen data.

Earlier (before 2019) state-of-the-art models often involved recurrent neural networks (Gröndahl et al., 2018).

For example, the CNN-GRU model by Zhang, Robinson & Tepper (2018) first extracts 2 to 4-gram features using convolutional layers with varying kernel sizes on word embeddings, then captures the sequence orders of these features with a gated recurrent unit (GRU) layer. This model outperformed previous models on six datasets when tested in-dataset. However, when tested cross-dataset by Gröndahl et al. (2018), the model's performance dropped even more than an LSTM, by over 30 points in macro-averaged F1.

Similarly, Badjatiya et al. (2017)'s model was once considered state-of-the-art when trained and evaluated on *Waseem*. Their two-stage training first produces word embeddings using a Long Short-Term Memory (LSTM) network through the same hate speech classification task, based on which another Gradient-Boosted Decision Tree (GBDT) classifier was trained. Arango, Prez & Poblete (2020) showed a similar F1 drop of around 30 points when applied on *HatEval*, and discussed a crucial methodological flaw—overfitting induced by extracting features on the combination of training and test set. Gröndahl et al. (2018) also reported that they failed to reproduce Badjatiya et al. (2017)'s results. Both Gröndahl et al. (2018) and Arango, Prez & Poblete (2020) also tested a Long Short-Term Memory (LSTM) network, which had been commonly used as a strong baseline. The performance drop was similar to the above two state-of-the-art models by Zhang, Robinson & Tepper (2018) and Badjatiya et al. (2017).

Since the introduction of BERT (Devlin et al., 2019), itself and its variants have been established as the new state-of-the-art. This is seen through the comparison to other neural networks (Swamy, Jamatia & Gambäck, 2019) and on the leaderboards of shared tasks, such as Zampieri et al. (2020); Fersini, Nozza & Rosso (2020). The general approach is to fine-tune a model, which had been pre-trained on domain-general data, on a target classification dataset. Yet, BERT and its variants are no exception to the lack of generalisation, although the cross-dataset performance drop is seemingly smaller. In cross-dataset experiments with four datasets, macro-averaged F1 scores decreased by 2 to 30 points (Swamy, Jamatia & Gambäck, 2019), which is less drastic compared to earlier state-of-the-art neural networks tested in other studies (Gröndahl et al., 2018; Arango, Prez & Poblete, 2020). Pamungkas, Basile & Patti (2020) and Fortuna, Soler-Company & Wanner (2021) also found that BERT and ALBERT tended to generalise the best across the models they experimented with.

Building upon BERT, a handful of recent studies suggest that additional hate-specific knowledge from outside the fine-tuning dataset might help with generalisation. Such knowledge can come from further masked language modelling pre-training on an abusive corpus (Caselli et al., 2021), or features from a hate speech lexicon (Koufakou et al., 2020).

Other models that have been studied include traditional machine learning models, such as character n-gram Logistic Regression (Gröndahl et al., 2018), character n-gram Multi-Layer Perceptron (MLP) (Gröndahl et al., 2018; Waseem, Thorne & Bingel, 2018), Support Vector Machines (Karan & Šnajder, 2018; Fortuna, Soler-Company & Wanner, 2021; Pamungkas & Patti, 2019; Pamungkas, Basile & Patti, 2020), and shallow networks with pre-trained embeddings, e.g., MLP with Byte-Pair Encoding (BPE)-based subword embeddings (Heinzerling & Strube, 2018; Waseem, Thorne & Bingel, 2018) and FastText (Joulin et al., 2017a; Wiegand, Ruppenhofer & Kleinbauer, 2019; Fortuna, Soler-Company & Wanner, 2021).

Generally, these simpler models do not perform as good as deep neural networks, such as LSTM (Pamungkas & Patti, 2019) and especially BERT and its variants (Pamungkas, Basile & Patti, 2020; Fortuna, Soler-Company & Wanner, 2021), in- or cross-dataset. However, exceptions exist in some dataset combinations, especially when it comes to

generalising. For example, n-gram Logistic Regression when comparing to LSTM (Gröndahl et al., 2018), SVM when comparing to LSTM and BERT (Pamungkas & Patti, 2019; Pamungkas, Basile & Patti, 2020), and FastText when comparing to BERT (Fortuna, Soler-Company & Wanner, 2021).

These cross-dataset studies only cover some of the more representative and/or recent hate speech detection models, but one can expect that the generalisation problem go beyond this small sample, and is far more ubiquitous in existing models than what these studies cover.

Despite the significance of the problem, systematic studies that compared a variety of models with datasets controlled are very limited (Arango, Prez & Poblete, 2020; Pamungkas & Patti, 2019; Pamungkas, Basile & Patti, 2020; Fortuna, Soler-Company & Wanner, 2021); there is also limited overlap in the datasets used between different studies (Table 2). Thus, one should be careful when drawing conclusions on the relative generalisability of models.

ii.Experimental setup

Using multiple hate speech datasets, we evaluated the accuracy of existing as well as our hate speech detection approaches.

iii.Data preprocessing and features.

For simplicity and generality, preprocessing and feature identification is intentionally minimal. For pre-processing, we apply case-folding, tokenization, and punctuation removal (while keeping emoji). For features, we simply extract word TF-IDF from unigram to 5-gram and character N-gram counts from unigram to 5-gram.

iv.Datasets.

We evaluate the approach on the Stormfront [14], TRAC [19], HatEval, and HatebaseTwitter [9] datasets previously described. These datasets provide a variety of hate speech definitions and aspects (including multiple types of aggression), and multiple types of online content (including online forums, Facebook, and Twitter content). For Stormfront, we use the balanced train/test split proposed in [14], with a random selection

of 10% of the training set held out as validation data. For the TRAC dataset, we use the English Facebook training, validation, and test splits provided by [19]. For Hateful, we use a split of the training set for validation and use the official validation dataset for testing because the official test set is not public. Finally, for the HatebaseTwitter dataset [9], we use the standard train-validation-test split provided by [9].

The use of pre-trained embeddings (discussed earlier) and parameter dropout (Srivastava et al., 2014) have been accepted as standard practice in the field of NLP to prevent over-fitting, and are common in hate speech detection as well. Nonetheless, the effectiveness of domain-general embedding models is questionable, and there has been only a limited number of studies that looked into the *relative* suitability of different pre-trained embeddings on hate speech detection tasks (Chen, McKeever & Delany, 2018; Mishra, Yannakoudakis & Shutova, 2018; Bodapati et al., 2019).

In Swamy, Jamatia & Gambäck (2019)'s study of model generalisability, abusive language-specific pre-trained embeddings were suggested as a possible solution to limited dataset sizes. Alatawi, Alhothali & Moria (2020) proposed White Supremacy Word2Vec (WSW2V), which was trained on one million tweets sourced through white supremacy-related hashtags and users. Compared to general word2vec (Mikolov et al., 2013) and GloVe (Pennington, Socher & Manning, 2014) models trained on news, Wikipedia, and Twitter data, WSW2V captured meaning more suitable in the hate speech context –e.g., ambiguous words like "race" and "black" have higher similarity to words related to ethnicity than sports or colours. Nonetheless, their WSW2V-based LSTM model did not consistently outperform Twitter GloVe-based LSTM model or BERT (Devlin et al., 2019). They did not consider cross-dataset testing for generalisablity, either.

The pre-training for BERT (and its variants) is both data and computationally-heavy, which limits the feasibility of training the hate speech equivalent of BERT from scratch. A reasonable compromise to that is performing further Masked Language-Modelling pre-training before the fine-tuning stage. By further pre-training RoBERTa (Liu et al., 2019), Wiedemann, Yimam & Biemann (2020) achieved first place at the Offenseval 2020 shared task (Zampieri et al., 2020). Caselli et al. (2021) pre-trained BERT further on

a larger-scale dataset of banned abusive subreddits and observed improvement over standard BERT on three Twitter datasets (*OLID, AbuseEval, HatEval*), in-dataset for all cases and cross-dataset for most cases. Both studies show that abusive language-specific pre-training, built upon generic pre-training, can be beneficial for both in-dataset performance and cross-dataset generalisation. The main downside is that the improvement gains, ranging from less than 1% to 4% in macro F1, seem disproportionate to the computational cost—Wiedemann, Yimam & Biemann (2020) only did the training on a small sample due to hardware limitations; it took Caselli et al. (2021) 18 days to complete 2 million training steps on one Nvidia V100 GPU. There also exists a trade-off between precision and recall for the positive class due to the domain shift (Caselli et al., 2021).

Research on transfer learning from other tasks, such as sentiment analysis, also lacks consistency. Uban & Dinu (2019) pre-trained a classification model on a large sentiment dataset (https://help.sentiment140.com/), and performed transfer learning on the *OLID* and *Kumar* datasets. They took pre-training further than the embedding layer, comparing word2vec (Mikolov et al., 2013) to sentiment embeddings and entire-model transfer learning. Entire-model transfer learning was found to be always better than using the baseline word2vec (Mikolov et al., 2013) model, but the transfer learning performances with only the sentiment embeddings were not consistent.

More recently, Cao, Lee & Hoang (2020) also trained sentiment embeddings through classification as part of their proposed model. The main differences are: the training data was much smaller, containing only *Davidson* and *Founta* datasets; the sentiment labels were produced by VADER (Gilbert & Hutto, 2014); their model was deeper and used general word embeddings (Mikolov et al., 2013; Pennington, Socher & Manning, 2014; Wieting et al., 2015) and topic representation computed through Latent Dirichlet Allocation (LDA) (Blei, Ng & Jordan, 2003) in parallel. Through ablation studies, they showed that sentiment embeddings were beneficial for both *Davidson* and *Founta* datasets.

Use of existing knowledge from a more mature research field like that of sentiment analysis has the potential to be used to jumpstart the relatively newer field of hate speech detection. It also offers a compromise between hate speech models, which might not be

generalisable enough, and completely domain-general models, which lack knowledge specific to hate speech detection. Nonetheless, more investigation into the conditions in which transfer learning works best to increase generalisability in particular still needs to be done.

Testing a model on a different dataset from the one which it was trained on is one way to more realistically estimate models' generalisability (Wiegand, Ruppenhofer & Kleinbauer, 2019). This evaluation method is called cross-dataset testing (Swamy, Jamatia & Gambäck, 2019) or cross-application (Gröndahl et al., 2018), and sometimes cross-domain classification (Wiegand, Ruppenhofer & Kleinbauer, 2019) or detection (Karan & Šnajder, 2018) if datasets of other forms of offensive language are also included.

As more hate speech and offensive language datasets emerged, a number of studies have touched upon cross-dataset generalisation since 2018, either studying generalisability per se, or as part of their dataset validation. The datasets they use (Table 1) to some extent reflect the best-known datasets in hate speech and other types of offensive language. These studies are further compared in Table 2 in terms of the models and datasets they used. As different datasets and models were investigated, instead of specific performance metrics, the remainder of this section will discuss the general findings of these studies, which can be roughly grouped into those on models and those on training and evaluation data.

CHAPTER 4

METHODOLOGY

Collecting and annotating data for the training of automatic classifiers to detect hate speech is challenging. Specifically, identifying and agreeing whether specific text is hate speech is difficult, as per previously mentioned, there is no universal definition of hate speech. Ross, et al. studied the reliability of hate speech annotations and suggest that annotators are unreliable [11]. Agreement between annotators, measured using Krippendorff's α, was very low (up to 0.29). However, they compared annotations based on the Twitter definition, versus annotations based on their own opinions and found a strong correlation.

Furthermore, social media platforms are a hotbed for hate speech, yet many have very strict data usage and distribution policies. This results in a relatively small number of datasets available to the public to study, with most coming from Twitter (which has a more lenient data usage policy). While the Twitter resources are valuable, their general applicability is limited due to the unique genre of Twitter posts; the character limitation results in terse, short-form text. In contrast, posts from other platforms are typically longer and can be part of a larger discussion on a specific topic. This provides additional context that can affect the meaning of the text.

Another challenge is that there simply are not many publicly-available, curated datasets that identify hateful, aggressive, and insulting text. A representative sampling of available training and evaluation public datasets is shown in Table 1:

- **HatebaseTwitter** [9]. One Twitter dataset is a set of 24,802 tweets provided by Davidson, et al [9]. Their procedure for creating the dataset was as follows. First they took a hate speech lexicon from Hatebase [16] and searched for tweets containing these terms, resulting in a set of tweets from about 33,000 users. Next they took a timeline from all these users resulting in a set of roughly 85 million Tweets. From the set of about 85 million tweets, they took a random sample, of 25k tweets, that contained terms from the lexicon. Via crowdsourcing, they annotated each tweet as hate speech, offensive (but not hate speech), or neither hate speech nor offensive. If the agreement between annotators was too low, the tweet was excluded from the set. A commonly-used subset of this dataset is also available,

containing 14,510 tweets.

- **WaseemA** [17]. Waseem and Hovy also provide a dataset from Twitter, consisting of 16,914 tweets labeled as racist, sexist, or neither [17]. They first created a corpus of about 136,000 tweets that contain slurs and terms related to religious, sexual, gender, and ethnic minorities. From this corpus, the authors themselves annotated (labeled) 16,914 tweets and had a gender studies major review the annotations.
- **WaseemB** [18]. In a second paper, Waseem creates another dataset by sampling a new set of tweets from the 136,000 tweet corpus [18]. In this collection, Waseem recruited feminists and anti-racism activists along with crowdsourcing for the annotation of the tweets. The labels therein are racist, sexist, neither or both.
- **Stormfront** [14]. de Gilbert, et al. provide a dataset from posts from a white supremacist forum, Stormfront [14]. They annotate the posts at sentence level resulting in 10,568 sentences labeled with Hate, NoHate, Relation, or Skip. Hate and NoHate labels indicate presence or lack thereof, respectively, of hate speech in each sentence. The label "Relation" indicates that the sentence is hate speech when it is combined with the sentences around it. Finally, the label "skip" is for sentences that are non-English or not containing information related to hate or non-hate speech. They also capture the amount of context (i.e., previous sentences) that an annotator used to classify the text.
- **TRAC** [19]. The 2018 Workshop on Trolling, Aggression, and Cyberbullying (TRAC) hosted a shared task focused on detecting aggressive text in both English and Hindi [19]. Aggressive text is often a component of hate speech. The dataset from this task is available to the public and contains 15,869 Facebook comments labeled as overtly aggressive, covertly aggressive, or non-aggressive. There is also a small Twitter dataset, consisting of 1,253 tweets, which has the same labels.
- **HatEval** [20]. This dataset is from SemEval 2019 (Task 5) for competition on multilingual detection of hate targeting to women and immigrants in tweets [20]. It consists of several sets of labels. The first indicates whether the tweet expresses hate towards women or immigrants, the second, whether the tweet is aggressive, and the third, whether the tweet is directed at an individual or an entire group. Note that targeting an individual is not necessarily considered hate speech by all definitions.
- **Kaggle** [21] Kaggle.com hosted a shared task on detecting insulting comments [21]. The dataset consists of 8,832 social media comments labeled as insulting or not insulting. While not necessarily hate speech, insulting text may indicate hate

speech.

- **GermanTwitter** [11]. As part of their study of annotator reliability, Ross, et al. created a Twitter dataset in German for the European refugee crisis [11]. It consists of 541 tweets in German, labeled as expressing hate or not.

We propose a multi-view SVM model for the classification of hate speech. It applies a multiple-view stacked Support Vector Machine (mSVM) [34]. Each type of feature (e.g., a word TF-IDF unigram) is fitted with an individual Linear SVM classifier (inverse regularization constant $C = 0.1$), creating a *view-classifier* for those features. We further combine the view classifiers with another Linear SVM ($C = 0.1$) to produce a *meta-classifier*. The features used in the meta-classifier are the predicted probability of each label by each view-classifier. That is, if we have 5 types of features (e.g., character unigram to 5-gram) and 2 classes of labels, 10 features would serve as input into the meta-classifier.

Combining machine learning classifiers is not a new concept [35]. Previous efforts have shown that combining SVM with different classifiers provides improvements to various data mining tasks and text classification [36, 37]. Combining multiple SVMs (mSVMs) has also been proven to be an effective approach in image processing tasks for reducing the large dimensionality problem [38].

However, applying multiple SVMs to identify hate speech expands the domain of use for such classification beyond that previously explored. Multi-view learning is known for capturing different *views* of the data [34]. In the context of hate speech detection, incorporating different views captures differing aspects of hate speech within the classification process. Instead of combining all features into a single feature vector, each view-classifier learns to classify the sentence based on only one type of feature. This allows the view-classifiers to pick up different aspects of the pattern individually.

Integrating all feature types in one model, by regularization, risks the masking of relatively weak but key signals. For example, “yellow” and “people” individually would appear more times than “yellow people” combined; posts having these terms individually are unlikely to be hate. However, “yellow people” is likely hate speech (especially when other hate speech aspects are present), but the signal might be rare in the collection, and therefore, is likely masked by the regularization if all features are combined together. In this case, mSVM is able to pick up this feature in one of the view-classifiers, where there

are fewer parameters.

Furthermore, this model offers the opportunity to interpret the model so as to identify which view-classifier contributes most through the meta-classifier provides human intuition for the classification. The view-classifier contributing most to the final decision identifies key vocabulary (features) resulting in a hate speech label. This contrasts with well-performing neural models, which are often opaque and difficult to understand [10, 39, 40]. Even state-of-the-art methods that employ self-attention (e.g., BERT [26]) suffer from considerable noise that vastly reduces interpretability.

Most if not all proposed hate speech detection models rely on supervised machine learning methods, where the ultimate purpose is for the model to learn the real relationship between features and predictions through training data, which generalises to previously unobserved inputs (Goodfellow, Bengio & Courville, 2016). The generalisation performance of a model measures how well it fulfils this purpose.

To approximate a model's generalisation performance, it is usually evaluated on a set-aside test set, assuming that the training and test data, and future possible cases come from the same distribution. This is also the main way of evaluating a model's ability to generalise in the field of hate speech detection.

Most of these studies only worked with English data. Yet, it is worth stressing that hate speech is a universal problem that exists in many languages, and generalisation studies focused on languages other than English are to date very sparse, despite the importance of the problem. Thus, research on cross-lingual generalisation is still in early stages.

One way to look at generalisation in non-English hate speech detection is applying the same cross-dataset evaluation on multiple datasets in another language. However, such studies do not yet exist. This is related to the fact that the majority of datasets are in English, which reflects linguistic and cultural unevenness in this field of research (Poletto et al., 2020; Vidgen & Derczynski, 2020).

Cross-lingual generalisation can be considered a more "extreme" type of generalisation (Arango, Prez & Poblete, 2020). The ideal case would be to be able to use data in one language for training and apply the model on data in another language, which would help address the challenge in low-resource languages. In a few studies (Pamungkas, Basile &

Patti, 2020; Glavaš, Karan & Vulić, 2020; Arango, Prez & Poblete, 2020; Fortuna, Soler-Company & Wanner, 2021), language was included as a separate variable, alongside a "domain" variable independent to it, which is characterised by the source platform or the data collection method.

Although these studies all touch on the same problem, how they evaluate cross-lingual performance differs. There are two main ways of enabling cross-lingual experiments: translating data and using multi-lingual models. These studies differ mainly by whether they perform translation on training or testing data and whether the translation is automatic or manual. Studies that use different evaluation methods also tend to look at the difficulty of the task differently. For example, Fortuna, Soler-Company & Wanner (2021) hold that multilingual generalisation per se is likely to be worse than its monolingual counterpart, while Arango, Prez & Poblete (2020) consider the two types of generalisation similar.

The factors that contribute to cross-lingual generalisation are similar to those in the monolingual setting as discussed above, with a few additional challenges:

- In terms of models, pre-trained multilingual word embeddings (MUSE (Conneau et al., 2017)) and language models (mBERT (Devlin et al., 2019), XLM-R (Conneau et al., 2020)) are frequently chosen as baselines. They are an intuitive and easily accessible starting point for cross-lingual experiments, but their limitations are also clear—the "curse of multilinguality" trades off single-language performance for its broad language coverage, as displayed in the results of the cross-lingual generalisation studies mentioned above (Pamungkas & Patti, 2019; Pamungkas, Basile & Patti, 2020; Glavaš, Karan & Vulić, 2020; Arango, Prez & Poblete, 2020; Fortuna, Soler-Company & Wanner, 2021) and in other tasks (Conneau et al., 2020). Similarly to the monolingual case, there are cases where traditional machine learning models outperform deep learning ones, such as SVM (Pamungkas, Basile & Patti, 2020) and GBDT (Arango, Prez & Poblete, 2020) compared to LSTM. Adding automatically translated training data alongside the original is beneficial (Pamungkas & Patti, 2019; Pamungkas, Basile & Patti, 2020).
- When it comes to the data, the most prominent additional factor compared to the monolingual setting is the similarity between the training (source) and

testing (target) languages. For instance, Among the wide range of languages that Glavaš, Karan & Vulić (2020) have tested, the cross-lingual performance drop between English, the source language, and German, the most similar target language, was less than one third of that between English and Turkish, when using mBERT on *Wulczyn*.

Although these studies more or less consider the "language" and "domain" variables as separate, there exists evidence that the two types of generalisation interact with each other. Studies that control the language variable more carefully tend to show a smaller drop across languages—for example, by manually translating exactly the same data (Glavaš, Karan & Vulić, 2020), as opposed to using automatic translation (Pamungkas & Patti, 2019; Pamungkas, Basile & Patti, 2020; Arango, Prez & Poblete, 2020) or different language dataset from the same shared task (Pamungkas & Patti, 2019; Pamungkas, Basile & Patti, 2020; Fortuna, Soler-Company & Wanner, 2021). Furthermore, adding data from a different domain can act as a regulariser from overfitting to the training language (Glavaš, Karan & Vulić, 2020).

As more datasets emerge, we can expect more generalisation studies considering language as a parameter in the near future. For the remainder of this paper, we discuss issues that can apply to hate speech detection in any language.

Obstacles to Generalisable Hate Speech Detection

Demonstrating the lack of generalisability is only the first step in understanding this problem. This section delves into three key factors that contribute to it: (1) presence of non-standard grammar and vocabulary, (2) paucity of and biases in datasets, and (3) implicit expressions of hate.

ii.Lexical Analyser

A number of previous works have attempted to generate sentiment words representing negative and positive orientation [14-16].The methods for generating opinion lexicon falls into two main categories, dictionary and corpus-based approaches. The former involves a static dictionary of semantically relevant words tagged with both a polarity label and semantic orientation score or reliability label [17-18]. The dictionary, in a

number of the proposed methods, is initially generated using a bootstrapping strategy that uses a small set of seed opinion words and an online dictionary such as WordNet [8] and SentiWordNet [19]. There exist substantial resources of dictionaries of opinion lexicon built from mainly adjectives, but also from verbs, adverbs and nouns [17, 20]. Esuli et al. in [19] uses a semisupervised method together with WordNet term relationships such as synonym, antonym and hyponymy to automatically generate a lexical resource that assigns each synset of WordNet three sentiment scores, summing up to one, regarding positivity, negativity, and objectivity, respectively. They leverage on a core seed of words that are known prior to carry a positive, negative or objective bias and iteratively add on new synsets using WordNet relations. Dictionary based approaches generally suffer from the inability to find opinion words with domain and context specific orientations. Corpus-based approaches use a domain corpus to capture opinion words with a preferred syntactic or cooccurrence patterns. Using natural language processing rule-based techniques, syntactic, structural and sentence level features are used in determining the semantic orientation of words and phrases to be included in an opinion lexicon. With this method, a lexicon is populated with words and phrases that are more attuned to t he domain by incorporating contextual features that could potentially change the semantic orientation of an opinion word. Features such as intensifiers could amplify or reduce the intensity of a neighboring lexicon item, while negations such as no, never, may change the directionality of a lexicon item. In [18] Wilson et al use a phrase-level sentiment analysis approach that first determines whether an expression is neutral or polar and then disambiguates the polarity of the polar expression.

iii. Subjectivity Analysis

To learn a subjectivity classifier both rule-based and learning-based approaches have been used [25]. Rule-based methods do not involve learning and typically rely on a precompiled list or a dictionary of subjectivity clues. Learning-based approaches use Machine language algorithms on both labeled and unlabeled corpora to learn patterns or other subjective clues. For the task of subjective sentence detection we employ a rule-based approach to classify sentences relying on a lexicon of well-established clues. In particular, we utilize two known sentiment lexicon resources of Wilson et al. [7, 26] and SentiWordNet [19]. The former comprises a list of over 8000 prior-polarity subjective

clues tagged with positive, negative, neutral and both tags. Besides, the clues also have a reliability tag that labels each clue as either strongly subjective (strongsubj) or weakly subjective (weaksubj). In subsequent references, we refer to this lexicon as SUBJCLUE. We use the widely employed criterion that considers a sentence as subjective if it contains two or more clues designated as strong subjective clues

Based on the definition of hate speech, in the following we describe the process of extracting and developing a semantic dictionary of hate domain features from our corpus. The semantics of hate not only include typical opinion words with negative and positive polarities, but also employ rich linguistic stylistic devices. From similes to metaphors to juxtapositions, haters are in no shortage of language to pass their nihilistic motives. While a number of documents include the direct use of incitement and violent words, others use less explicit expressions and may be inexplicable if we rely only on opinion-oriented words. For example, rival groups are compared to concepts such as beasts, crocodiles and grasshoppers. The rule-based hate speech classifier that we ultimately create relies on three different sets of features. In the following we explain the features and the rules used to develop them. Negative Polarity From the subjective sentences identified through the process in section V-1, we identify opinionated words that have a negative semantic orientation. All the word features in our corpus that match reliability tag of either weakly or strongly subjective and a polarity tag of negative in the SUBJ CLUE lexicon are extracted and included in our polarity features lexicon. Hate verbs As our second set of features we include “hate” verbs that are not part of the SUBJCLUE lexicon. The idea is to extract all the verbs that bear a relation with hate verbs from our hate corpora. Based on the definition of hate speech in section II, common in hate speech is the use of terms that condone and encourage violence acts. Such terms include the verbs discriminate, loot, riot, beat, kill, and evict. Beginning with an initial seed list of the six verbs, we use bootstrapping and WordNet’ssynsets, and hypernym relationships to build the list from all the verbs in the hate corpus and include as our hate lexicon only those verbs that overlap with our corpus but are not in the SUBJCLUE. If a verb in the seed list matches any of the verbs in the corpus list of verbs, then it is included in the lexicon list. For each round of iteration, we build a new seed list by reinitializing it with occurrences of unique words from the corpus list of verbs.

CHAPTER 5

CODING AND TESTING

```
import string

import pandas as pd

import numpy as np

from sklearn.feature_extraction.text import CountVectorizer

from sklearn.model_selection import train_test_split

from sklearn.tree import DecisionTreeClassifier

import nltk

import re

from re import sub

from nltk.corpus import stopwords

from sklearn.metrics import accuracy_score


stopwords = (stopwords.words("english"))

stemmer = nltk.PorterStemmer()

data = pd.read_csv("labeled_data.csv")


data["labels"] = data["class"].map({0: "Hate Speech", 1: "Offensive Speech", 2:
"Neither"})
```

```
data = data[["tweet", "labels"]]



def cleaning_text(text):
    text = str(text).lower()
    text = sub('[.?]', '', text)
    text = sub('https?://\S+|www.\S+', '', text)
    text = sub('<.?>+', '', text)
    text = sub(r'[^\w\s]', '', text)
    text = sub('/n', '', text)
    text = sub('\w\d\w', '', text)
    text = [word for word in text.split(" ") if word not in stopwords]
    text = " ".join(text)
    text = [stemmer.stem(word) for word in text.split(" ")]
    text = " ".join(text)
    return text



data["tweet"] = data["tweet"].apply(cleaning_text)



x = np.array(data["tweet"])
```

```
y = np.array(data["labels"])

cv = CountVectorizer()

X = cv.fit_transform(x)

X_train,X_test,y_train,y_test = train_test_split(X,y,test_size=0.33, random_state=42)
model = DecisionTreeClassifier()
model.fit(X_train,y_train)

y_pred = model.predict(X_test)

print(accuracy_score(y_test,y_pred))

example = "nigga you are so dark that we have to search for you even in the daylight"
example = cv.transform([example]).toarray()
print(model.predict(example))
```

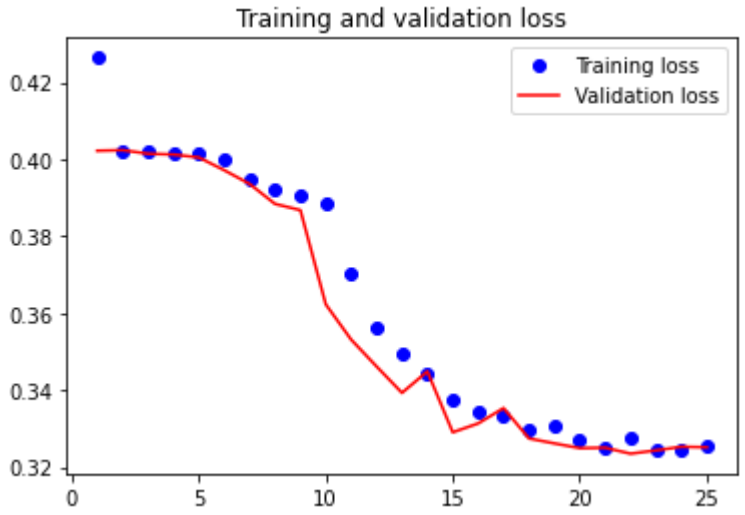

Fig 1.0 Loss graph

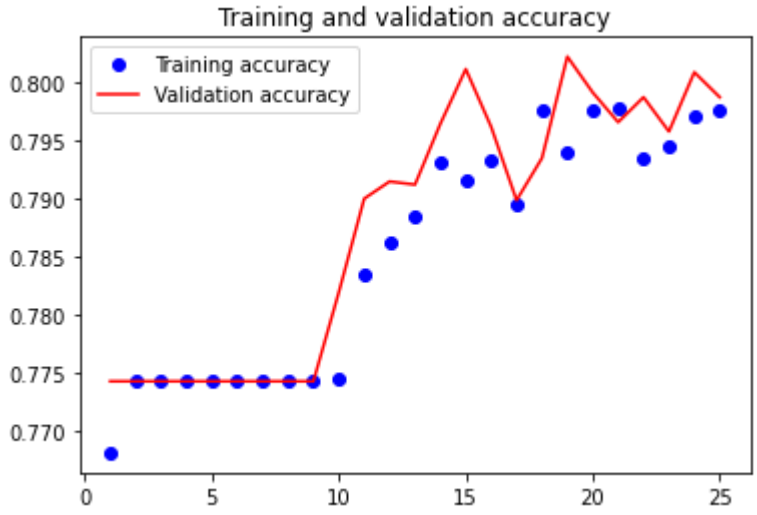

Fig 1.1 Accuracy graph

CHAPTER 6

SCREENSHOTS AND RESULTS

i.The cross-lingual case

Training data has a pronounced influence on generalisation. The performance drops in models highlight the differences in the distribution of posts between datasets (Karan & Šnajder, 2018), yet some datasets are more similar to each other. Furthermore, certain attributes of a dataset could lead to more generalisable models.

Similarity between datasets varies, as there are groups of datasets that produce models that test much better on each other. For example, in Wiegand, Ruppenhofer & Kleinbauer (2019)'s study, FastText models (Joulin et al., 2017a) trained on three datasets *(Kaggle, Founta, Razavi)* achieved F1 scores above 70 when tested on one another, while models trained or tested on datasets outside this group achieved around 60 or less. In Swamy, Jamatia & Gambäck (2019)'s study with fine-tuned BERT models (Devlin et al., 2019), *Founta* and *OLID* produced models that performed well on each other. The source of such differences are usually traced back to search terms (Swamy, Jamatia & Gambäck, 2019), topics covered (Nejadgholi & Kiritchenko, 2020; Pamungkas, Basile & Patti, 2020), label definitions (Pamungkas & Patti, 2019; Pamungkas, Basile & Patti, 2020; Fortuna, Soler-Company & Wanner, 2021), and data source platforms (Glavaš, Karan & Vulić, 2020; Karan & Šnajder, 2018).

Another way of looking at generalisation and similarity is by comparing differences between individual classes across datasets (Nejadgholi & Kiritchenko, 2020; Fortuna, Soler & Wanner, 2020; Fortuna, Soler-Company & Wanner, 2021), as opposed to comparing datasets as a whole. In both Nejadgholi & Kiritchenko (2020) and Fortuna, Soler-Company & Wanner (2021)'s experiments, the best generalisation is achieved for more general labels such as "toxicity", "offensive", or "abusive". Generalisation is not as good for finer-grained hate speech labels. All in all, these findings are indicative of an imbalance of the finer-grained subclasses, particularly owing to disagreements in the definition of what constitutes hate speech, which proves more difficult than defining what

constitutes offensive language.

ii.Non-standard grammar and vocabulary

Within the hate speech labels, the relative similarity also varies. Fortuna, Soler & Wanner (2020) used averaged word embeddings (Bojanowski et al., 2017; Mikolov et al., 2018) to compute the representations of classes from different datasets, and compared classes across datasets. One of their observations is that *Davidson*'s "hate speech" is very different from *Waseem*'s "hate speech", "racism", "sexism", while being relatively close to *HatEval*'s "hate speech" and *Kaggle*'s "identity hate". This echoes with experiments that showed poor generalisation of models from *Waseem* to *HatEval* (Arango, Prez & Poblete, 2020) and between *Davidson* and *Waseem* (Waseem, Thorne & Bingel, 2018; Gröndahl et al., 2018).

The proportion of abusive posts in a dataset, first of all, plays a part. Swamy, Jamatia & Gambäck (2019) holds that a larger proportion of abusive posts (including hateful and offensive) leads to better generalisation to dissimilar datasets, such as *Davidson*. This is in line with Karan & Šnajder (2018)'s study where *Kumar* and *Kolhatkar* generalised best, and Waseem, Thorne & Bingel (2018)'s study where models trained on *Davidson* generalised better to *Waseem* than the other way round. In contrast, in Wiegand, Ruppenhofer & Kleinbauer (2019)'s study, the datasets with the least abusive posts generalised the best (*Kaggle* and *Founta*). Similarly, Fortuna, Soler-Company & Wanner (2021) could not confirm the impact of class proportions. Nejadgholi & Kiritchenko (2020) offered an explanation to this: there exists a trade-off between true positive and true negative rates dictated by the class proportions, which impacts the minority class performance the most but this is not always reflected in the overall F1 score.

iii.Hate speech, offensive language, and abusive language

In terms of what properties of a dataset lead to more generalisable models, there are frequently mentioned factors, but also inconsistency across different studies. Interactions between factors, which contribute to the inconsistency, are also reported.

Biases in the samples are also frequently mentioned. Wiegand, Ruppenhofer & Kleinbauer (2019) hold that less biased sampling approaches produce more generalisable models. This was later reproduced by Razo & Kübler (2020) and also helps explain their results with the two datasets that have the least positive cases. Similarly, Pamungkas & Patti (2019) mentioned that a wider coverage of phenomena lead to more generalisable models. So do topics that are more general rather than platform-specific (Nejadgholi & Kiritchenko, 2020).

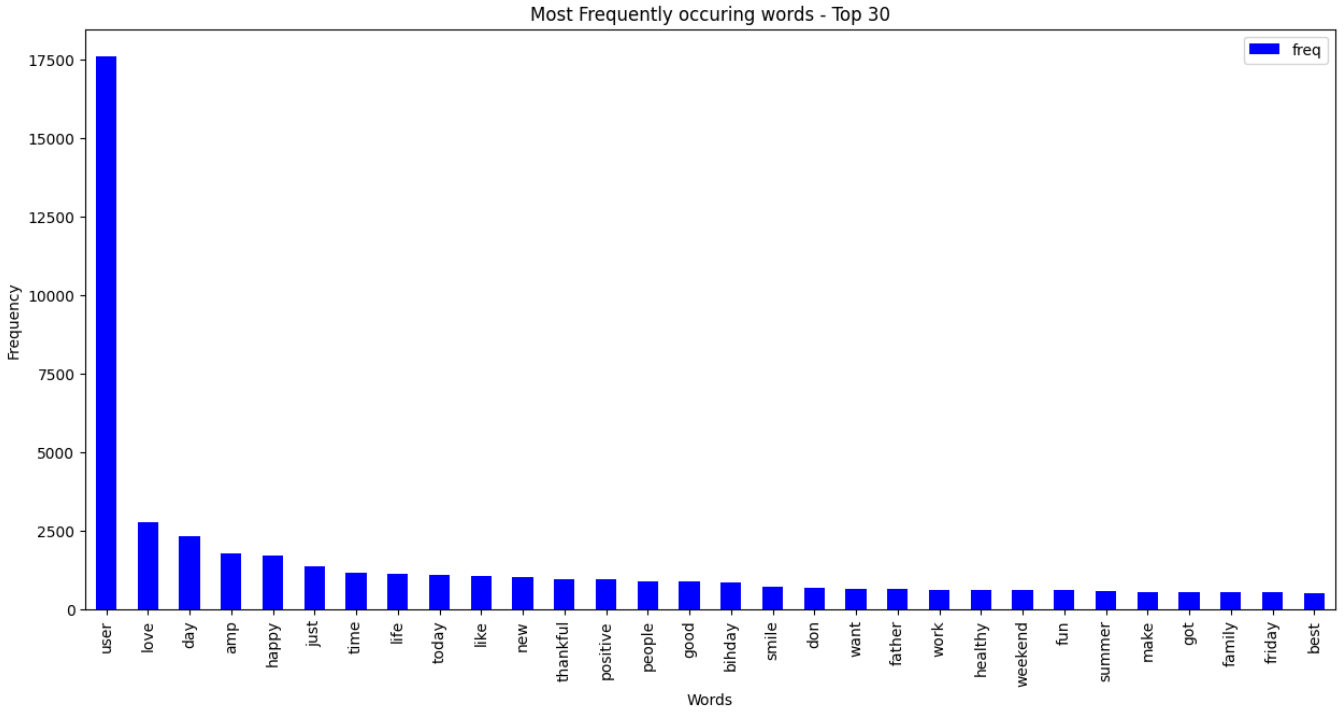

Fig 2.0 Frequently occurring words graph

A larger training data size is generally believed to produce better and more generalisable models (Halevy, Norvig & Pereira, 2009). It is mentioned as one of the two biggest factors contributing to cross-dataset performance in Karan & Šnajder (2018)'s study. Caselli et al. (2020) also found that, on *HatEval*, their dataset (*AbuseEval*) produced a model even better-performing than the one trained on *HatEval* end-to-end. They partially attributed this to a bigger data size, alongside annotation quality. However, the benefit of having more data is counterbalanced by data distribution differences (Karan & Šnajder, 2018), as discussed above. Moreover, its relative importance compared to other factors seems to be small, when the latter are carefully controlled (Nejadgholi & Kiritchenko, 2020; Fortuna, Soler-Company & Wanner, 2021).

From a domain-specific perspective, Taylor, Peignon & Chen (2017) and Magu & Luo (2018) attempted to identify code words for slurs used in hate communities. Both of them used keyword search as part of their sourcing of Twitter data and word embedding models to model word relationships. Taylor, Peignon & Chen (2017) identified hate communities through Twitter connections of the authors of extremist articles and hate speech keyword searches. They trained their own dependency2vec (Levy & Goldberg, 2014) and FastText (Bojanowski et al., 2017) embeddings on the hate community tweets and randomly sampled "clean" tweets, and used weighted graphs to measure similarity and relatedness of words. Strong and weak links were thus drawn from unknown words to hate speech words. In contrast, Magu & Luo (2018) collected potentially hateful tweets using a set of known code words. They then computed the cosine similarity between all words based on a word2vec model (Mikolov et al., 2013) pre-trained on news data. Code words, which have a neutral meaning in news context, were further apart from other words which fit in the hate speech context. Both Taylor, Peignon & Chen (2017) and Magu & Luo (2018) focused on the discovery of such code words and expanding relevant lexicons, but their methods could potentially complement existing hate lexicons as classifier features or for data collection.

Recently, an increasing body of research is approaching the problem by adapting character or sequence-level features to evade the challenge posed by words:

The benefit of character-level features has not been consistently observed. Three studies compared character-level, word-level, and hybrid (both character- and word-level) CNNs, but drew completely different conclusions. Park (2018) and Meyer & Gambäck (2019) found hybrid and character CNN to perform best respectively. Probably most surprisingly, Lee, Yoon & Jung (2018) observed that word and hybrid CNNs outperformed character CNN to similar extents, with all CNNs performing worse than character n-gram logistic regression. Small differences between these studies could have contributed to this inconsistency. More importantly, unlike the word components of the models, which were initialized with pre-trained word embeddings, the character embeddings were trained end-to-end on the very limited respective training datasets. It is thus likely that these character embeddings overfit on the training data.

iv. Generalizability in hate speech detection

In contrast, simple character n-gram logistic regression has shown results as good as sophisticated neural network models, including the above CNNs (Van Aken et al., 2018; Gao & Huang, 2017; Lee, Yoon & Jung, 2018). Indeed, models with fewer parameters are less likely to overfit. This suggests that character-level features themselves are very useful, when used appropriately. A few studies used word embeddings that were additionally enriched with subword information as part of the pre-training. For example, FastText (Bojanowski et al., 2017) models were consistently better than hybrid CNNs (Bodapati et al., 2019). In addition, a MIMICK (Pinter, Guthrie & Eisenstein, 2017)-based model displayed similar performances (Mishra, Yannakoudakis & Shutova, 2018).

The use of sentence embeddings partially solves the out-of-vocabulary problem by using the information of the whole post instead of individual words. Universal Sentence Encoder (Cer et al., 2018), combined with shallow classifiers, helped one team (Indurthi et al., 2019) achieve first place at the HatEval 2019 shared task (Basile et al., 2019). Sentence embeddings, especially those trained with multiple tasks, also consistently outperformed traditional word embeddings (Chen, McKeever & Delany, 2019).

Large language models with sub-word information have the benefits of both subword-level word embeddings and sentence embeddings. They produce the embedding of each word with its context and word form. Indeed, BERT (Devlin et al., 2019) and its variants have demonstrated top performances at hate or abusive speech detection challenges recently (Liu, Li & Zou, 2019; Mishra & Mishra, 2019).

Nonetheless, these relatively good solutions to out-of-vocabulary words (subword- and context-enriched embeddings) all face the same short-coming: they have only seen the standard English retrieved from BookCorpus and Wikipedia. NLP tools perform best when trained and applied in specific domains (Duarte, Llanso & Loup, 2018). In hate speech detection, word embeddings trained on relevant data (social media or news sites) had a clear advantage (Chen, McKeever & Delany, 2018; Vidgen et al., 2020). The domain mismatch could have similarly impaired the subword- and context-enriched

models' performances. There is little work so far on adapting them to the abusive domain to increase model generalisability so far (Caselli et al., 2021).

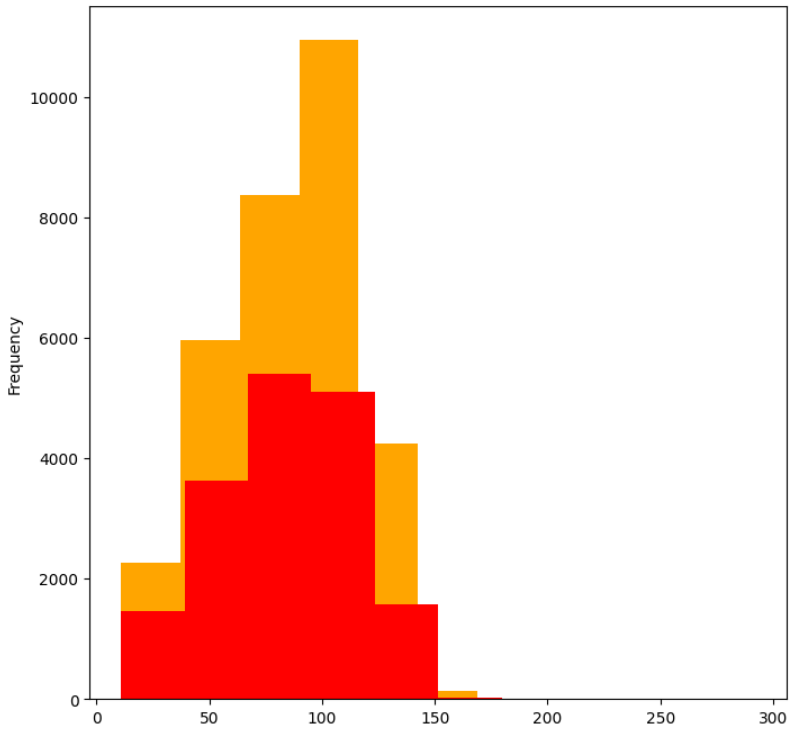

Fig 2.1 Frequency graph

Hate speech detection, which is largely focused on social media, shares similar challenges to other social media tasks and has its specific ones, when it comes to the grammar and vocabulary used. Such user language style introduces challenges to generalisability at the data source, mainly by making it difficult to utilise common NLP pre-training approaches.

On social media, syntax use is generally more casual, such as the omission of punctuation (Blodgett & O'Connor, 2017). Alternative spelling and expressions are also used in dialects (Blodgett & O'Connor, 2017), to save space, and to provide emotional emphasis (Baziotis, Pelekis & Doulkeridis, 2017). Sanguinetti et al. (2020) provided extensive guidelines for studying such phenomena syntactically.

Commonly seen in hate speech, the offender adopts various approaches to evade content moderation. For example, the spelling of offensive words or phrases can be obfuscated (Nobata et al., 2016; Serrà et al., 2017), and common words such as "Skype", "Google",

and “banana” may have a hateful meaning—sometimes known as euphemism or code words (Taylor, Peignon & Chen, 2017; Magu & Luo, 2018).

When the spelling is obfuscated, a word is considered out-of-vocabulary and thus no useful information can be given by the pre-trained models. In the case of code words, pre-trained embeddings will not reflect its context-dependent hateful meaning. At the same time, simply using identified code words for a lexicon-based detection approach will result in low precision (Davidson et al., 2017). As there are infinite ways of combining the above alternative rules of spelling, code words, and syntax, hate speech detection models struggle with these rare expressions even with the aid of pre-trained word embeddings.

v. Limited, biased labelled data.

In practice, this difficulty is manifested in false negatives. Qian et al. (2018) found that rare words and implicit expressions are the two main causes of false negatives; Van Aken et al. (2018) compared several models that used pre-trained word embeddings, and found that rare and unknown words were present in 30% of the false negatives of Wikipedia data and 43% of Twitter data. Others have also identified rare and unknown words as a challenge for hate speech detection (Nobata et al., 2016; Zhang & Luo, 2018). More recently, Fortuna, Soler-Company & Wanner (2021) drew a more direct line between out-of-vocabulary words and generalization performance, by showing that the former is one of the top contributing features in a classifier for the latter. It has also been shown as an important factor in the cross-lingual case (Pamungkas, Basile & Patti, 2020).

Obstacles to generalizability also lie in dataset construction, and dataset size is the relatively most unequivocal one. When using machine learning models, especially deep learning models with millions of parameters, small dataset size can lead to overfitting and in turn harm generalizability (Goodfellow, Bengio & Courville, 2016).

It is particularly challenging to acquire labelled data for hate speech detection as knowledge or relevant training is required of the annotators. As a high-level and abstract concept, the judgement of “hate speech” is subjective, needing extra care when processing annotations. Hence, datasets are usually not big in size.

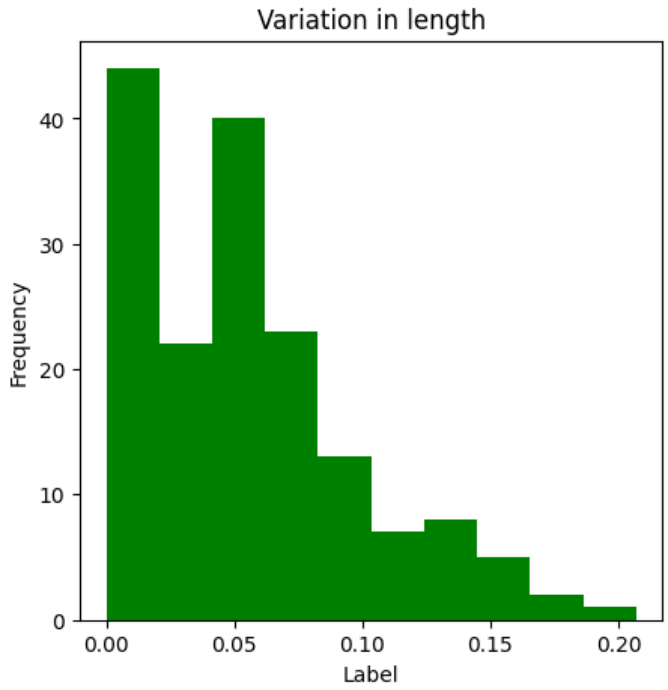

Fig 2.3 Variation in length with respect to frequency graph

Obstacles to generalisability also lie in dataset construction, and dataset size is the relatively most unequivocal one. When using machine learning models, especially deep learning models with millions of parameters, small dataset size can lead to overfitting and in turn harm generalisability (Goodfellow, Bengio & Courville, 2016).

Across different stages, Google Scholar was the main search engine, and two main sets of keywords were used. References and citations were checked back-and-forth, with the number of iterations depending on how coarse or fine-grained the search of that stage was.

- General keywords: “hate speech”, “offensive”, “abusive”, “toxic”, “detection”, “classification”.
- Generalisation-related keywords: “generalisation” (“generalization”), “generalisability” (“generalizability”), “cross-dataset”, “cross-domain”, “bias”.

We started with a pre-defined set of keywords. Then, titles of proceedings of the most relevant recent conferences and workshops (Workshop on Abusive Language Online, Workshop on Online Abuse and Harms) were skimmed, to refine the set of keywords. We also modified the keywords during the search stages as we encountered new phrasing of the terms. The above keywords shown are the final keywords.

Main literature search stages

Before starting to address the aims of this paper, an initial coarse literature search involved searching for the general keywords, skimming the titles and abstracts. During this stage, peer-reviewed papers with high number of citations, published in high-impact venues were prioritised. Existing survey papers on hate speech and abusive language detection (Schmidt & Wiegand, 2017; Fortuna & Nunes, 2018; Al-Hassan & Al-Dossari, 2019; Mishra, Yannakoudakis & Shutova, 2019; Vidgen et al., 2019; Poletto et al., 2020; Vidgen & Derczynski, 2020) were also used as seed papers. The purpose of this stage was to establish a comprehensive high-level view of the current state of hate speech detection and closely related fields.

Feature sets	FIRST Corpus			SECOND Corpus		
	Precision	Recall	F-score	Precision	Recall	F-score
Semantic	67.21	66.23	66.72	66.34	65.62	66.03
Semantic +hate	71.22	68.23	70.69	70.14	67.90	69.00
Semantic+hate+theme-based	73.42	68.42	70.83	71.55	68.24	69.85

Table1.Classification Results for Strongly Hateful Sentences

For the first aim of this paper—building a comparative summary of existing research on generalisability in hate speech detection—the search mainly involved different combinations of the general and generalisation-related keywords. As research on this topic is sparse, during this stage, all papers found and deemed relevant were included.

Building upon the first two stages, the main obstacles towards generalisable hate speech detection were then summarised: (1) presence of non-standard grammar and vocabulary, (2) paucity of and biases in datasets, and (3) implicit expressions of hate. This was done through extracting and analysing the error analysis of experimental studies found in the first stage, and comparing the results and discussions of the studies found in the second stage. Then, for each category of obstacles identified, another search was carried out, involving combinations of the description and paraphrases of the challenges and the general keywords. The search in this stage is the most fine-grained, in order to ensure coverage of both the obstacles and existing attempts to address them. After the main

search stages, the structure of the main findings in the literature was laid out. During writing, for each type of findings, the most representative studies were included in the writing up. We defined the relative representativeness within studies we have found, based on novelty, experiment design and error analysis, publishing venues, and influence. We also prioritised studies that addressed problems specific to hate speech, compared to better-known problems that are shared with other offensive language and social media tasks.

It is particularly challenging to acquire labelled data for hate speech detection as knowledge or relevant training is required of the annotators. As a high-level and abstract concept, the judgement of "hate speech" is subjective, needing extra care when processing annotations. Hence, datasets are usually not big in size.

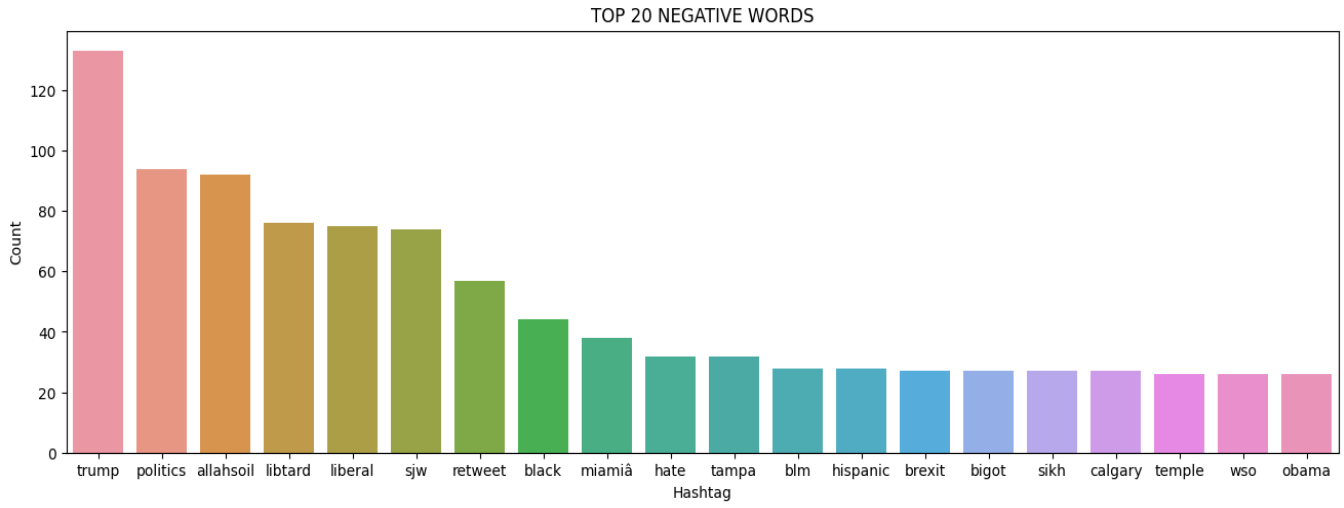

Fig 2.4 Top 20 negative words graph

In addition to a limited size, datasets are also prone to biases. Non-random sampling and subjective annotations introduce individual biases, and the different sampling and annotation processes across datasets further increase the difficulty of training models that can generalise across heterogeneous data.

Hate speech and, more generally, offensive language generally represent less than 3% of social media content (Zampieri et al., 2019b; Founta et al., 2018). To alleviate the effect of scarce positive cases on model training, all existing social media hate speech or offensive content datasets used boosted (or focused) sampling with simple heuristics.

This compares the sampling methods of hate speech datasets studied the most in

cross-dataset generalisation. Consistently, keyword search and identifying potential hateful users are the most common methods. However, what is used as the keywords (slurs, neutral words, profanity, hashtags), which users are included (any user from keyword search, identified haters), and the use of other sampling methods (identifying victims, sentiment classification) all vary a lot.

Feature sets	FIRST Corpus			SECOND Corpus		
	Precision	Recall	F-score	Precision	Recall	F-score
Semantic	58.42	61.12	59.73	56.68	57.54	57.11
Semantic +hate	63.24	64.42	63.82	61.56	62.24	61.90
Semantic+hate+theme-based	65.32	64.92	65.12	63.78	64.00	63.89

Table2.Classification Results for Strongly Hateful Sentences without Using Subjective Sentences

Trying to make use of the general assumption that hateful messages contain specific negative words (such as slurs, insults, etc.), many authors utilize the presence of such words as a feature. To obtain this type of information lexical resources are required that contain such predictive expressions. A popular source for such word lists is the web. There are several publicly available lists that consist of general hate-related terms.3 Apart from works that employ such lists (Xiang et al., 2012; Burnap and Williams, 2015; Nobata et al., 2016), there are also approaches, such as Burnap and Williams (2016) which focus on lists that are specialized towards a particular subtype of hate speech, such as ethnic slurs4 , LGBT slang terms5 , or words with a negative connotation towards handicapped people.6 Apart from publicly-available word lists from the web other approaches incorporate lexicons that have been specially compiled for the task at hand. Spertus (1997) employs a lexicon comprising so-called good verbs and good adjectives. Razavi et al. (2010) manually compiled an Insulting and Abusing Language Dictionary containing both words and phrases with different degrees of manifestation of flame varieties. This dictionary also assigns weights to each lexical entry which represents the degree of the potential impact level for hate speech detection.

CHAPTER 7

CONCLUSION AND FUTURE ENHANCEMENTS

i. Conclusion

In this paper, we investigated the application of deep neural network architectures for the task of hate speech detection. We found them to significantly outperform the existing methods. Embeddings learned from deep neural network models when combined with gradient boosted decision trees led to best accuracy values. In the future, we plan to explore the importance of the user network features for the task.

As hate speech continues to be a societal problem, the need for automatic hate speech detection systems becomes more apparent. We presented the current approaches for this task as well as a new system that achieves reasonable accuracy. We also proposed a new approach that can outperform existing systems at this task, with the added benefit of improved interpretability. Given all the challenges that remain, there is a need for more research on this problem, including both technical and practical matters.

Hate speech detection is a difficult task to accomplish because it involves processing text and understanding the context. The hate speech data sets are usually not clean, so they need to be pre-processed before classification algorithms can detect hate speech in them. Different machine learning models have different strengths that make some better than others for certain tasks such as detecting hate speech. Some models are more accurate while others are more efficient. It is important to use different models and compare their performance in order to find the best one for hate speech detection. Pre-training methods have become popular in recent years and it is important to test whether they work well with hate speech detection algorithms. It is also important to see how hate speech detection models can be used to address domain changes.

ii. Future work

In this paper, we presented a survey on the automatic detection of hate speech. This task is usually framed as a supervised learning problem. Fairly generic features, such as bag of words or embeddings, systematically yield reasonable classification performance. Character-level approaches work better than token-level approaches. Lexical resources, such as list of slurs, may help classification, but usually only in combination with other types of features. Various complex features using more linguistic knowledge, such as dependencyparse information, or features modelling specific linguistic constructs, such as imperatives or politeness, have also been shown to be effective. Information derived from text may not be the only cue suggesting the presence of hate speechFuture work would consider evaluation of ensemble methods on additional test sets (e.g. SemEval 2014 and 2015 for example). Also, a comparison of different weighting schemes is likely useful to understand variations within this parameter. Beyond that, building models with different network configurations and embedding models are all considered to be natural next steps. Different approaches, such as LSTM networks based on character representations (as opposed to word embeddings) should be considered. Reproducing the promising results using LSTM and Gradient Boosted Decision Trees (Badjatiya et al., 2017) on additional datasets is a worthwhile exercise too. Given knowledge that neural network performance improves as datasets become larger, it would be an interesting experiment to gain insight as to what amount of data is sufficient enough where ensemble methods do not provide a boost in performance. Therefore one possible next step for our work would be to try our methods on progressively larger datasets to empirically show that ensembles provide smaller improvements as training data increases.

8. REFERENCES

[1] Al-Hassan A, Al-Dossari H. 2019. Detection of hate speech in social networks: a survey on multilingual corpus. In: Computer Science & Information Technology (CS & IT). Chennai, India. AIRCC Publishing Corporation. 83-100

[2] Alatawi HS, Alhothali AM, Moria KM. 2020. Detecting white supremacist hate speech using domain specific word embedding with deep learning and BERT.

[3] Alorainy W, Burnap P, Liu H, Williams ML. 2019. The enemy among us: detecting cyber hate speech with threats-based othering language embeddings. *ACM Transactions on the Web* 13(3):1-26

[4] Arango A, Pérez J, Poblete B. 2020. Hate speech detection is not as easy as you may think: a closer look at model validation (extended version) *Information Systems* Epub ahead of print 2020 30 June

[5] Badjatiya P, Gupta M, Varma V. 2019. Stereotypical bias removal for hate speech detection task using knowledge-based generalizations. In: Liu L, White RW, Mantrach A, Silvestri F, McAuley JJ, Baeza-Yates R, Zia L, eds. The World Wide Web conference, WWW 2019, San Francisco, CA, USA, May 13-17, 2019. ACM. 49-59

[6] Badjatiya P, Gupta S, Gupta M, Varma V. 2017. Deep learning for hate speech detection in tweets. In: Proceedings of the 26th international conference on World Wide Web companion. 759-760

[7] Banko M, MacKeen B, Ray L. 2020. A unified taxonomy of harmful content. In: Proceedings of the fourth workshop on online abuse and harms. Association for Computational Linguistics. 125-137

[8] Basile V. 2020. It's the end of the gold standard as we know itOn the impact of pre-aggregation on the evaluation of highly subjective tasks. In: CEUR workshop proceedings. 10

[9] Basile V, Bosco C, Fersini E, Nozza D, Patti V, Rangel Pardo FM, Rosso P, Sanguinetti M. 2019. SemEval-2019 Task 5: multilingual detection of hate speech against immigrants and women in twitter. In: Proceedings of the 13th international workshop on semantic evaluation. Minneapolis, Minnesota, USA. Association for Computational Linguistics. 54-63

[10] Baziotis C, Pelekis N, Doulkeridis C. 2017. DataStories at SemEval-2017 Task 4: Deep LSTM with attention for message-level and topic-based sentiment analysis. In: Proceedings of the 11th international workshop on semantic evaluation (SemEval-2017). Vancouver, Canada. Association for Computational Linguistics. 747-754

[11] Blei DM, Ng AY, Jordan MI. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research* 3(Jan):993-1022

[12] Blodgett SL, Green L, OConnor B. 2016. Demographic dialectal variation in social media: a case study of African-American English. In: Proceedings of the 2016 conference on empirical methods in natural language processing. 1119-1130

[13] Blodgett SL, O'Connor B. 2017. Racial disparity in natural language processing: a case study of social media African-American English.

[14] Bodapati S, Gella S, Bhattacharjee K, Al-Onaizan Y. 2019. Neural word decomposition models for abusive language detection. In: Proceedings of the third workshop on abusive language online. Florence, Italy. Association for Computational Linguistics. 135-145

[15] Bojanowski P, Grave E, Joulin A, Mikolov T. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics* 5:135-146

[16] Bolukbasi T, Chang K, Zou JY, Saligrama V, Kalai AT. 2016. Man is to computer programmer as woman is to homemaker? Debiasing word embeddings. In: Lee DD, Sugiyama M, von Luxburg U, Guyon I, Garnett R, eds. Advances in neural information processing systems 29: Annual conference on neural information processing systems 2016, December 5-10, 2016, Barcelona, Spain. 4349-4357

[17] Breitfeller L, Ahn E, Jurgens D, Tsvetkov Y. 2019. Finding microaggressions in the wild: a case for locating elusive phenomena in social media posts. In: Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP). Hong Kong, China. Association for Computational Linguistics. 1664-1674

[18] Buolamwini J, Gebru T. 2018. Gender shades: Intersectional accuracy disparities in commercial gender classification. In: Conference on fairness, accountability and transparency. 77-91

[19] Cao R, Lee RK.-W, Hoang T.-A. 2020. DeepHate: hate speech detection via multi-faceted text representations. In: 12th ACM conference on web science, WebSci '20. New York, NY, USA. Association for Computing Machinery. 11-20

[20] Caruana R. 1997. Multitask learning. *Machine Learning* 28(1):41-75

[21] Caselli T, Basile V, Mitrović J, Kartoziya I, Granitzer M. 2020. I feel offended, don't be abusive! implicit/explicit messages in offensive and abusive language. In: Proceedings of the 12th language resources and evaluation conference. Marseille, France. European Language Resources Association. 6193-6202

[22] Caselli T, Basile V, Mitrovi J, Granitzer M. 2021. HateBERT: retraining BERT for abusive language detection in english.

[23] Cer D, Yang Y, Kong S-y, Hua N, Limtiaco N, StJohn R, Constant N, Guajardo-Cespedes M, Yuan S, Tar C+2 more. 2018. Universal sentence encoder for english. In: Proceedings of the 2018 conference on empirical methods in natural language processing: system demonstrations. Brussels, Belgium. Association for Computational Linguistics. 169-174

[24] Chen H, McKeever S, Delany SJ. 2018. A comparison of classical versus deep learning techniques for abusive content detection on social media sites. In: Staab S, Koltsova O, Ignatov DI, eds. *Social informatics*, Lecture notes in computer science. Cham: Springer International Publishing. 117-133

[25] Chen H, McKeever S, Delany SJ. 2019. The use of deep learning distributed representations in the identification of abusive text. *Proceedings of the International AAAI Conference on Web and Social Media* 13:125-133

[26] Chung Y-L, Kuzmenko E, Tekiroglu SS, Guerini M. 2019. CONAN - Counter narratives through nichesourcing: a multilingual dataset of responses to fight online hate speech. In: Proceedings of the 57th annual meeting of the association for computational linguistics. Florence, Italy. Association for Computational Linguistics. 2819-2829

[27] Cohen J. 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement* 20(1):37-46

[28] Conneau A, Khandelwal K, Goyal N, Chaudhary V, Wenzek G, Guzmán F, Grave É, Ott M, Zettlemoyer L, Stoyanov V. 2020. Unsupervised cross-lingual representation learning at scale. In: Proceedings of the 58th annual meeting of the association for computational linguistics. 8440-8451

[29] Conneau A, Lample G, Ranzato M, Denoyer L, Jégou H. 2017. Word translation without parallel data.

[30] Daumé III H. 2007. Frustratingly easy domain adaptation. In: Proceedings of the 45th annual meeting of the association of computational linguistics. 256-263

[31] Davidson T, Bhattacharya D, Weber I. 2019. Racial bias in hate speech and abusive language detection datasets. In: *Proceedings of the third workshop on abusive language online*. Florence. Association for Computational Linguistics. 25-35

[32] Davidson T, Warmsley D, Macy M, Weber I. 2017. Automated hate speech detection and the problem of offensive language.

[33] De Gibert O, Perez N, García-Pablos A, Cuadros M. 2018. 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. 11-20

[34] Devlin J, Chang M-W, Lee K, Toutanova K. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 conference of the north American chapter of the association for computational linguistics: human language technologies, Volume 1 (Long and Short Papers). 4171-4186

[35] Dixon L, Li J, Sorensen J, Thain N, Vasserman L. 2018. Measuring and mitigating unintended bias in text classification. In: Proceedings of the 2018 AAAI/ACM conference on AI, ethics, and society. New Orleans LA USA. ACM. 67-73

[36] Duarte N, Llanso E, Loup A. 2018. Mixed messages? The limits of automated social media content analysis. In: Conference on fairness, accountability and transparency. 106

[37] Fersini E, Nozza D, Rosso P. 2018. Overview of the Evalita 2018 task on automatic misogyny identification (AMI) In: Caselli T, Novielli N, Patti V, Rosso P, eds. *EVALITA evaluation of NLP and speech tools for Italian*. Accademia University Press. 59-66

[38] Fersini E, Nozza D, Rosso P. 2020. AMI @ EVALITA2020: automatic misogyny identication. In: Proceedings of the 7th evaluation campaign of Natural Language Processing and Speech tools for Italian (EVALITA 2020). 8

[39] Fersini E, Rosso P, Anzovino M. 2018. Overview of the Task on Automatic Misogyny Identication at IberEval 2018. In: Proceedings of the third workshop on evaluation of human language technologies for iberian languages (IberEval 2018).

Seville, Spain. 214-228

[40] Fortuna P, Nunes S. 2018. A survey on automatic detection of hate speech in text. *ACM Computing Surveys* 51(4):1-30

[41] Fortuna P, Soler J, Wanner L. 2020. Toxic, hateful, offensive or abusive? what are we really classifying? An empirical analysis of hate speech datasets. In: Proceedings of the 12th language resources and evaluation conference. Marseille, France. European Language Resources Association. 6786-6794

[42] Fortuna P, Soler-Company J, Wanner L. 2021. How well do hate speech, toxicity, abusive and offensive language classification models generalize across datasets? *Information Processing & Management* 58(3):102524

[43] Founta A-M, Djouvas C, Chatzakou D, Leontiadis I, Blackburn J, Stringhini G, Vakali A, Sirivianos M, Kourtellis N. 2018. Large scale crowdsourcing and characterization of Twitter abusive behavior. In: Proceedings of ICWSM. AAAI Press.

[44] Gao L, Huang R. 2017. Detecting online hate speech using context aware models. In: Proceedings of the international conference recent advances in natural language processing, RANLP 2017. Varna. INCOMA Ltd. 260-266

[45] Gao L, Kuppersmith A, Huang R. 2017. Recognizing explicit and implicit hate speech using a weakly supervised two-path bootstrapping approach. In: Proceedings of the eighth international joint conference on natural language processing (Volume 1: Long Papers). Taipei, Taiwan. Asian Federation of Natural Language Processing. 774-782

[46] Gilbert C, Hutto E. 2014. Vader: a parsimonious rule-based model for sentiment analysis of social media text. In: Eighth international conference on weblogs and social media (ICWSM-14), volume 81. 82

[47] Agrawal S, Awekar A. 2018. Deep learning for detecting cyberbullying across multiple social media platforms. In: European conference on information retrieval. Grenoble, France. Springer. 141-153

Github Profile Links:

Geetha Shashank Pericherla- https://github.com/Shashank-Pericherla?tab=repositories

Siddharth S - https://github.com/Sid20000/Hate_Speech_Detection

Om Tiwari - https://github.com/iamomtiwari/AI