Alma Mater Studiorum · Università di Bologna

DIPARTIMENTO DI INTERPRETAZIONE E TRADUZIONE
Corso di Laurea magistrale in Specialized Translation (classe LM-94)

TESI DI LAUREA
in NATURAL LANGUAGE PROCESSING

This Is My Cope: Identification and Forecasting of Hate Speech in Inceldom

CANDIDATO:
Paolo Gajo

RELATORE:
Alberto Barrón Cedeño
CORRELATRICE:
Silvia Bernardini
CORRELATORE:
Adriano Ferraresi

Anno Accademico 2022/2023
Primo Appello

Acknowledgements

I wish to thank my supervisor, Alberto Barrón Cedeño, for all the help and support he has given me not just during the writing of this thesis, but also in writing two papers (one of which was accepted, I owe you a beer), and for his availability and capability of fostering my interest in doing research in the field of NLP.

I also wish to extend my gratitude to my co-supervisors, Prof. Silvia Bernardini for initially working with me on the topic of this thesis, and Prof. Adriano Ferraresi, for his kind availability in answering my questions with regard to embarking on a PhD.

Thank you so much to Dave and Leo, it is hard to convey how much I value the friendships we have built in the past two years. Thanks to Alice and Rachele for not making me feel like the only STEM weirdo at the DIT. Thanks to Sasha and Julius for all the nice meals and conversations. Thanks to Alessandra, Edoardo, Francesca, Giada, Greta, Ilaria, Letizia, Marcell, Matilda, Monica, Seva, and Virginia: I will never forget all the chats and beers we had at Valverde and everywhere else around Forlì. Thanks to Vicky, Robi, Lesley, and Hafsa for letting me inhale some of their second-hand smoke every now and then.

Thank you, Martina, for giving me a shoulder to lean on during this arduous journey. I love you.

Last but not least, thanks Mom, for being an unwavering force of nature and for being a role model of commitment and dedication.

Abstract

The identification and moderation of hate speech on social media platforms is a crucial endeavour, which has the potential to increase the civility of online interactions and safeguard the well-being of all users. Despite the topic having been thoroughly explored in recent years by the NLP community, many avenues of research are still open, especially in the context of niche communities, where the language used by speakers is often riddled with opaque jargon and for which the amount of available data is limited. For the first time, we introduce a multilingual corpus for the analysis and identification of hate speech in the domain of inceldom, i.e., online spaces frequented by incels, short for "involuntary celibates". The corpus is built from incel web forums in English and Italian, including expert annotation at the post level for two kinds of hate speech: misogyny and racism. This resource paves the way for the development of mono- and cross-lingual models for (a) the identification of hateful (misogynous and racist) posts and (b) the forecasting of the amount of hateful responses that a post is likely to trigger. As regards the identification tasks, our experiments aim at improving the performance of Transformer models using masked language modeling (MLM) pre-training and dataset merging. These approaches are particularly effective in cross-lingual scenarios. Using multilingual MLM, we are able to improve the performance of mBERT models on the task of identifying hate speech in a zero-shot cross-lingual scenario by 17 points in terms of F_1-measure, while the performance boost is 34 and 18 points for misogyny and racism identification, respectively. Multilingual dataset merging also leads to a large performance increase for the binary classification setting, in the cross-lingual scenario, with a performance boost over the baseline dataset we compiled of 22 points in terms of F_1-measure, for the best MLM pre-trained model. In the forecasting setting, we propose a simple and novel approach to the task, which allows us to beat our MSE baseline by 37% in the monolingual setting.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Disclaimer: Due to the nature of the topic, this work contains highly offensive words.

Hate speech, broadly defined as language that expresses hatred towards a targeted group or is intended to be derogatory, humiliating, or insulting to its members (Davidson et al., 2017) has become an increasingly prevalent and dangerous phenomenon in the past years (Matamoros-Fernández and Farkas, 2021). The rapid rise of social media platforms has enabled the dissemination of hateful and offensive rhetoric, with tangible negative consequences, such as increased prejudice towards minority groups and the escalation of hate crimes (Pelicon et al., 2021).

A specific area of concern is represented by the conglomerate of online spaces known as the *Manosphere*, where misogynous discourse in particular has become increasingly rampant (Ribeiro et al., 2021). These spaces are characterized by the adoption of the *Red Pill* philosophy, which promotes a toxic idea of masculinity and traditional gender roles, and has been linked to the rise in misogynous and racist discourse (Ging, 2019). Specifically, the incel (short for "involuntary celibate") community within the Manosphere has been identified as one that frequently engages in hateful, misogynous, and racist speech (Nagle, 2017; Jaki et al., 2019).

Given the gravity of the phenomenon, especially in these environments, the development of effective hate speech detection systems is critical to addressing the harmful consequences of these online platforms and promoting a more inclusive and respectful digital landscape.

1.1 Contributions

With the aim of making social media platforms safer and more civil environments, we focus on hindering the spread of incel hate speech through the Internet. To this end, we attempt to answer the following research questions:

Q_1: Can we build new resources which can be used for the effective automatic detection of incel hate speech, both in monolingual and cross-lingual scenarios?

Q_2: Can we adapt existing automatic hate speech detection systems to the detection of incel hate speech and improve their performance on this task?

Q_3: Given a social media thread t of posts $p_i \in t$ with $i \in \{1, \ldots, n\}$ in which the first post is indicated as p', can we predict the number of hateful replies p' will receive?

In order to answer these research questions, in this thesis we present three macro contributions:[1]

C_1: We compile and release two new unsupervised corpora containing posts extracted from two incel forums, *Incels.is* and *Il forum dei brutti*, one in English and one in Italian, respectively. A subset of each was annotated with a binary label for misogyny and one for racism, which we combine to obtain hate speech labels. The unsupervised datasets can be used for language modeling, while the supervised datasets can be used for downstream fine-tuning for hate speech, misogyny and racism detection.

C_2: Using these resources, we approach the task of identifying hate speech (binary classification), and racism and misogyny (multi-label classification) within the domain of inceldom, in monolingual (English) and cross-lingual (from English to Italian) scenarios. This entails predicting whether a post from the aforementioned forums expresses hate speech in general in the binary classification setting, or misogyny and/or racism in the multi-label classification setting. We experiment with a variety of Transformer models. We

[1] All of the code and data used in this thesis is available at `https://github.com/paolo-gajo/Incel-Hate-Speech`

pre-train BERT models using the masked language modeling (MLM) task on the unsupervised datasets. In the binary classification setting, we further attempt improving models by fine-tuning them on combinations of our supervised datasets and combinations of existing datasets annotated for hate speech. Our experiments show that these approaches are effective, particularly in the cross-lingual scenarios, in which we obtain a 20-point F_1-measure increase for the binary task, and a 33-point and 18-point increase for the misogyny and racism detection tasks, respectively.

C_3: By leveraging the resources we have built and the newly obtained models, we attempt to answer Q_3. That is, we verify whether we can outperform a mean squared error (MSE) baseline in predicting the share of hateful replies the first post of a social media thread p' will receive. To do this, we automatically label our unsupervised datasets with the new improved models and use them to create datasets for regression training. We then use these *hate score* datasets to train Transformer models for regression. We show that in the monolingual scenario our Transformer model, pre-trained with MLM on our English unsupervised dataset, is capable of predicting with reasonable effectiveness the number of hateful replies p_i which p' will receive solely using its textual content, surpassing the MSE baseline by 37%.

1.2 Thesis Structure

The rest of the thesis is articulated in the following chapters:

Chapter 2 provides an overview of general theoretical concepts whose grasp is fundamental in order to fully understand the research subject at hand. We first present the sociolinguistic background of the research, introducing the concepts of hate speech, the so-called "Manosphere", incels and Internet forums. We also provide a brief explanatory section on corpus linguistics, which we use for part of our study. We then lay out the most important elements of deep learning, including basic neural networks, Transformers (Vaswani et al., 2017) and BERT (Devlin et al., 2019) models. We also introduce the metrics used to evaluate the performance of the models.

Chapter 3 presents past research specifically on incels and hate speech, both from a sociological and computational perspective. As far as natural language processing is concerned, we provide an overview of the most used

methods and resources for hate speech detection, along with the most recent developments in with regard to its forecasting on social media platforms.

Chapter 4 presents a corpus-driven study of incel language, conducted on *Incels.is* and *Il forum dei brutti*. We first describe the framework and theory of the study, which uses the concept of *keyness* to identify the most characteristic words of the incel language and then present the unsupervised datasets obtained by crawling and scraping the two forums, along with the results of the study.

Chapter 5 presents the framework of this study and the experiments conducted within it. A description of the annotation process of the unsupervised datasets is first provided, which allows us to present the experimental settings we approach. We then present three tasks: (*i*) binary classification of hate speech, (*ii*) multi-label classification of misogyny and racism, and (*iii*) hate speech forecasting. For each task, we first present the experimental setting and the adopted methodology, and then report the results obtained.

Chapter 6 provides a summary of the thesis, draws conclusions from the results obtained and discusses future work.

1.3 Research Products

The work presented in this thesis has been accepted at RANLP 2023[2] in the form of a short paper:

Gajo, P., Muti A., Korre K., Bernardini S., Barrón-Cedeño A. On the Identification and Forecasting of Hate Speech in Inceldom. In *Proceedings of the 2023 International Conference on Recent Advances in Natural Language Processing (RANLP 2023)*, 2023.

[2] `https://ranlp.org/ranlp2023/`

Chapter 2

Background

This chapter is concerned with introducing a variety of concepts whose understanding is necessary in order to fully grasp this study. Section 2.1 presents a number of fundamental concepts linked to the sociological and linguistic aspects of the research. Section 2.2 introduces neural networks, the model architectures used in the study and the sort of natural language processing tasks which can be approached with them. Section 2.3 introduces the Transformer model architecture. Section 2.4 presents BERT, the main model used in the experiments of this study. Section 2.5 introduces the metrics used to evaluate the performance of the models.

2.1 Sociolinguistic Concepts

This section provides background information on sociolinguistic aspects which the reader should be familiar with to gain a comprehensive understanding of the topics being discussed.

2.1.1 Hate Speech

Hate speech can be generally defined as "language that is used to express hatred towards a targeted group or is intended to be derogatory, to humiliate, or to insult the members of the group" (Davidson et al., 2017). By this definition, misogynous and racist posts on social media can be taken as examples of hateful speech. Indeed, misogyny can be defined as "the manifestation of hostility towards women because they are women" (Jurasz and Barker, 2019),

while (individual) racism[1] can be thought of as "beliefs in the superiority of one's race [...] characterized by 'behavioral enactments' between individuals that maintain a power differential between racial groups" (Neblett, 2019).

Hate speech is a dangerous phenomenon which can lead to poor psychological well-being, hate crime, and increased prejudice towards minority groups in both virtual and local communities (Pelicon et al., 2021). The prevalence of offensive language and hate speech on social media platforms has resulted in significant and concrete negative consequences. For example, unfettered online hate has resulted in an increase in anti-Muslim sentiment during former U.S. president Donald Trump's campaign, and against Rohingya Muslims in Myanmar in 2017 (Pelicon et al., 2021).

In the literature, the term "hate speech" has been grouped with other terms, such as "abusive language" and "harmful speech" (Waseem et al., 2017). However, the difference between hate speech and offensive language is crucial. As noted by Davidson et al. (2017), not all instances of offensive language can be considered hate speech, as certain terms might be offensive to a certain group, without necessarily constituting hate speech. That is, if a person from a certain group uses a word that could generally be considered as hateful if used toward that group, then it might not be considered hate speech. Conversely, were it said from an outsider to that group, then that would be more likely to be considered hate speech. It is therefore important to distinguish between these two categories of speech, as hate speech constitutes a more specific sub-category of language that is broadly offensive.

In this study, we consider hate speech as being speech that intends to discriminate a group (or an individual belonging to it) based on innate characteristics of their members. We adopt this definition based on the one provided by Davidson et al. (2017), also taking into account the definition given by Nockleby (2000), who defines hate speech as:

> "any communication that disparages a person or a group on the basis of some characteristic such as race, color, ethnicity, gender, sexual orientation, nationality, religion, or other characteristic."

[1] In this work, we only take into account instances of racist behavior at the individual level, not at the institutional level.

2.1.2 The Red Pill and the Manosphere

In the past years, misogynous discourse has been ever more prevalent in online spaces (Ribeiro et al., 2021). This trend is linked to the rise of the anti-feminist world view of the *Red Pill*,[2] a trending "philosophy" which claims to free men from a perceived feminist delusion which hinders them (Ging, 2019). Anonymity on platforms like Twitter and Reddit has allowed the number of hateful posts to increase dramatically, with women being more targeted than ever (Muti et al., 2022b). This paints the picture of a situation that is seemingly only getting worse in terms of hate speech, especially with regard to misogynous discourse spreading online.

Spaces which adhere to this philosophy are characterized by a toxic idea of masculinity (Dignam and Rohlinger, 2019). Influencers in these spaces peddle the idea that men and women can be grouped along an objective scale of value, based on certain characteristics. Often, such characteristics have to do with complying with a traditional view of gender roles (Freeman, 2020), as opposed to the supposedly misleading narratives pushed by more progressive ideologies. For example, according to this world view, high-value men should display typically masculine behavior, work towards having a good career, and take good care of their physique (Latimore, n.d.). Women, on the other hand, are expected to fulfill traditionally feminine roles, e.g., taking care of children and doing house chores. These ideologies have become especially pervasive in the past few years by spreading on social media platforms, through influencers who have sometimes gained worldwide popularity by promoting misogynous ideas (Das, 2022).

Globally, the conglomerate of online spaces that spreads hateful discourse in adherence with this set of beliefs is colloquially known as the *Manosphere* (Ging, 2019). Although the general purported goal of the communities making up this macro-environment is often to simply advocate for sensible matters in the broad context of men's rights or to discuss issues men face in society, in actuality they are more often than not breeding grounds for misogynous and racist discourse (Farrell et al., 2019; Ging, 2019).

[2]The term "Red Pill" originates from the 1999 film "The Matrix" where the protagonist is offered a red pill to reveal an unpleasant truth about the world (a constructed reality), and a blue pill to continue living in blissful ignorance. In this context, it represents a supposed awakening to a reality in which men are oppressed by feminism.

2.1.3 Incels

Among the communities which inhabit the Manosphere, one in particular has proved to convey especially hateful messaging through their extremely misogynous and racist discourse: incels, short for "involuntary celibates" (Nagle, 2017). Incels as a community mostly comprise men who are unsuccessful in finding a sexual partner. As such, they often consider women to be the source of their problems, venting their frustration on them, often through egregious expressions of hate (Jaki et al., 2019).

Incels place sexual relations within a strict framework of value they call the "sexual market" (Segalewitz, 2020). Within this system, they rate people on various scales of value, usually ranging from 1 to 10 (Gothard, 2020). Women, which they consider to be their counterparts in the sexual market, are usually objectified and labeled based on their beauty. For example, they can be rated on a scale and objectified with labels such as "Stacy" (with a rating of 9)[3] or "Becky" (with a rating of 5-7),[4] among other similarly misogynous classifications (Gothard, 2020).

In the incel worldview, women are seen as fundamentally immature people who commit *hypergamy* (Young, 2019). That is, they consider most women to only be sexually and romantically attracted to a small percentage of men. In addition, hypergamy as a concept underlies the idea that women "trade up", aiming for men who, on average, are more "valuable" than them on the sexual market. This results in a supposed system in which, according to incels, the top 20% of men, in terms of promiscuity, have 80% of the sexual encounters in society. However, as actual statistics show, this is not the case, with the top 20% having around 50-60% of the encounters (Stone, 2018).

Alongside blaming women for their shortcomings, incels frequently put substantial emphasis on the importance of "looks, money, and status", simply abbreviated as "LMS" (Young, 2019), i.e., what a man should supposedly possess in order to be able to have sexual relations. Incels also blame their inability to find a partner on their physical appearance (Gothard, 2020). The most frequent physical aspects they discuss are face conformation and body type, especially as regards height. This obsession with the topic has lead the community to develop very specific concepts to describe undesirable qualities, such as "hunter" vs. "prey" eyes.[5] Incels decry their physical flaws as a curse

[3] `https://incels.wiki/w/Stacy`
[4] `https://incels.wiki/w/Becky`
[5] `https://incels.wiki/w/Hunter_eyes`

in current society, considering it to be a death sentence as far as the possibility of finding a partner goes. This is particularly true within circles who adhere to the philosophy of the *Black Pill*, an even more extreme ideology, compared to the Red Pill, describing inceldom as a systemic problem which cannot be solved through personal effort alone (Glace et al., 2021).

Misogyny and racism are the two most common types of hate speech in these spaces (Silva et al., 2016; Ging and Siapera, 2018; Jaki et al., 2019) and they are often expressed in novel and unique ways. While it is obvious that women-hating men would be known for their misogynous speech, the same cannot be said about racism, which seems to be a curious component of the matter. Arguably, this phenomenon can be traced back to the fact that some of these spaces are tightly linked to the alt-right movement (Hoffman et al., 2020b).

The language and behavior of incels are particularly problematic due to the risk of radicalization they pose. The most famous example of incel ideology bringing about real-world harm is probably the mass killings perpetrated by Elliot Rodger in 2014, in Isla Vista (Jaki et al., 2019). Such incidents are concrete proof of the severity of the phenomenon and its potential to cause physical harm, besides the spread of discriminatory discourse it promotes online.

2.1.4 Internet Forums

The online spaces which incels use to communicate are varied, and include platforms such as Reddit (Gothard, 2020), Discord (Hoffman et al., 2020a) and conventional Internet forums (Pelzer et al., 2021). According to Holtz et al. (2012) an Internet forum, also known as a message board, is a virtual platform for online discussions. Typically, forums are organized in a tree-like structure, with various topics discussed within thematic sections and subsections. Users can initiate a conversation or "thread" by making a new post (known as the "original post" of the thread and abbreviated as "OP") within these sections, while others can respond to the OP or other users' comments by leaving a post.

An important difference between mainstream social media platforms, such as Reddit, and niche Internet forums is that the latter usually "fly under the radar" and do not garner much outside attention. Mainstream platforms

have been under legislative scrutiny in the past years[6] with more and more pressure being applied to them in order to curb toxic and dangerous phenomena developing and spreading within the spaces they manage. Not being under the spotlight allows smaller platforms to operate free from external influence, meaning that moderation is more lax, since there is no pressure from external third parties to regulate the content being posted. As far as this study is concerned, the most important consequence of this is that the language is consequently more genuine, and at times more extreme.

2.1.5 Corpus Linguistics

According to Baker (2010), corpus linguistics is a branch of linguistics which utilizes computer software to analyze extensive collections of electronically stored texts, while McEnery and Wilson (2003) characterize it as a "methodology". The analytical methods employed by this discipline rely on real-world instances of language, which allow us to derive rules and explore trends in a text (Baker, 2010). These techniques are usually applied on large quantities of text, which helps us notice underlying behavioral patterns that might otherwise go unnoticed.

One important feature which can be analyzed through these methods is the frequency with which words appear in language. Not all words are used at the same rate: for example, words we use every day, such as "pen" or "walk", will be used much more often than "inference" or "backpropagation". Usage frequency depends on various factors, such as the communicative context: a particular situation might require the use of lexicon typical of a specialized language (Baroni and Bernardini, 2004), which creates diastratic variation (Coseriu, 1981). For example, a conversation between lawyers discussing work matters will likely contain legal terms at a much higher frequency than one taking place between two random persons in a bar.

We can investigate the difference of relative frequency in word usage between general language and the language used in a specific speech community by building corpora representative of the two groups of speakers. That is, we can use a large *reference corpus*, representing general language usage, and compare its frequencies to a *focus corpus* (Kilgarriff, 2009), built only from texts pertaining to a specific communicative context.

[6] `https://www.brookings.edu/blog/techtank/2020/09/21/the-push-for-content-moderation-legislation-around-the-world/`

One metric which allows us to study this difference in frequency is the formulation of *keyness* used by Kilgarriff (2009), which indicates what words are highly frequent compared to a reference corpus. The keyness $k(w)$ of a word w is defined as (Lexical Computing Ltd., 2015):

$$k(w) = \frac{fpm_f(w) + n}{fpm_r(w) + n} \tag{2.1}$$

where fpm represents the normalized frequency of a word per million words, $fpm_f(w)$ refers to the frequency of the word in the focus corpus, $fpm_r(w)$ refers to the word in the reference corpus, and n is a smoothing parameter (we use the default, $n = 1$). $k(w) > 1$ indicates that the word is more frequent in the focus corpus than in the reference corpus, while $k(w) < 1$ indicates the opposite. The higher the value of $k(w)$, the greater the prevalence in frequency of the word in the focus corpus compared to the reference corpus.

2.2 Deep Learning

Deep learning is a subfield of machine learning, which in turn is a subfield of artificial intelligence (AI). Broadly speaking, AI is a field of science which aims to automate intellectual work that would otherwise be done by humans (Chollet, 2018, p. 4). To learn about data, i.e., about real world patterns, deep learning uses artificial *neural networks*. Inspired by biological neurons, these computational models are designed to recognize patterns in data and make decisions based on those patterns (Lane et al., 2019, p. 156). They are made up of various interconnected neurons, generally organized in layers, as shown in Figure 2.1.

Before data can be processed by a neural network, it must be converted into an appropriate format. This means converting certain characteristics about the input data, also known as *features* (Chollet, 2018, p. 18), into a numerical representation that can be fed into the first layer of the network. In the context of natural language processing (NLP), a field concerned with applying artificial intelligence to language tasks, examples of features include, besides the intrinsic characteristics of the text itself (e.g., its syntax and semantics), the number of words in a sentence, the name of its author, or the metadata associated with the author's account, in case of online messages (Sansonetti et al., 2020). The features are then fed into the first layer of the network, the *input layer*. The input layer is followed by one or more *hidden*

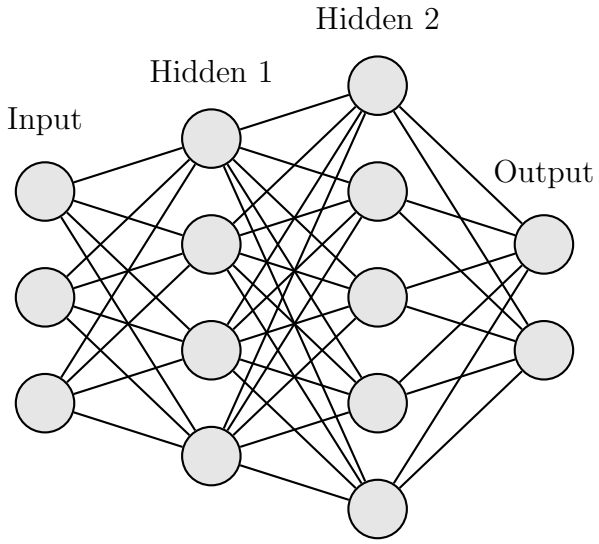


Figure 2.1: Diagram of a fully connected neural network with an input layer, two dense hidden layers and an output layer.

layers, which elaborate the information and pass the result to the final layer, the *output layer*, which provides a final output for the entire neural network.

The nodes that make up a neural network are also called *perceptrons* (Lane et al., 2019, p. 157), algorithms that mimic the operation of living neuron cells, receiving input signals from other neurons and "firing" whenever the received signal is strong enough. As shown in Figure 2.2, a perceptron receives an input array $\mathbf{x}$ of dimensionality n:

$$\mathbf{x} = [x_1, x_2, \ldots, x_n] \tag{2.2}$$

which is modified by a weight array $\mathbf{w}$ of the same dimensionality:

$$\mathbf{w} = [w_1, w_2, \ldots, w_n] \tag{2.3}$$

through a dot product operation:

$$\mathbf{x} \cdot \mathbf{w} = \sum_{i=1}^{n} x_i w_i \tag{2.4}$$

A bias b is then added to the result, obtaining the weighted sum z:

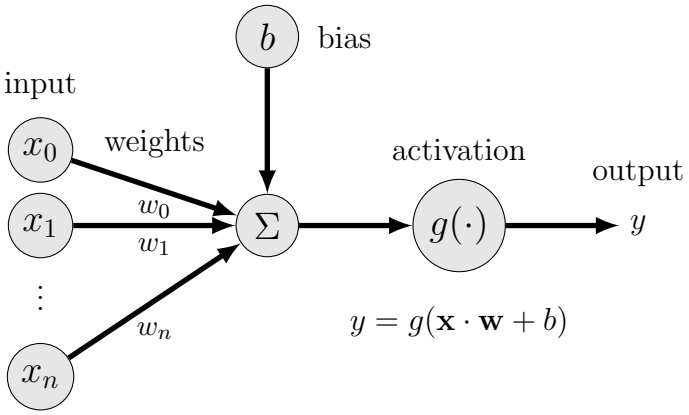


Figure 2.2: The diagram of a perceptron.

$$z = \mathbf{x} \cdot \mathbf{w} + b \tag{2.5}$$

which is then passed as a parameter to a function g, also known as the *activation function*, to calculate the output value $y = g(z)$. This process is also known as the *forward pass*, since the input is passed forward through the network until it reaches the output layer.

2.2.1 Activation Functions

There are many different types of activation functions, each with its own advantages and disadvantages. Some of the most common activation functions are the sigmoid function, the rectified linear unit (ReLU) function and the gaussian error linear unit (GELU) function. Figure 2.3 shows the three functions.

Sigmoid The sigmoid function is a smooth and continuous function that maps any real-valued input to a value between 0 and 1:

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{2.6}$$

It is widely used in binary classification tasks because its output falls in a range $0 \leq \sigma(x) \leq 1$ and can thus be interpreted as the probability of the positive class. However, sigmoids cannot be used in deep neural networks to

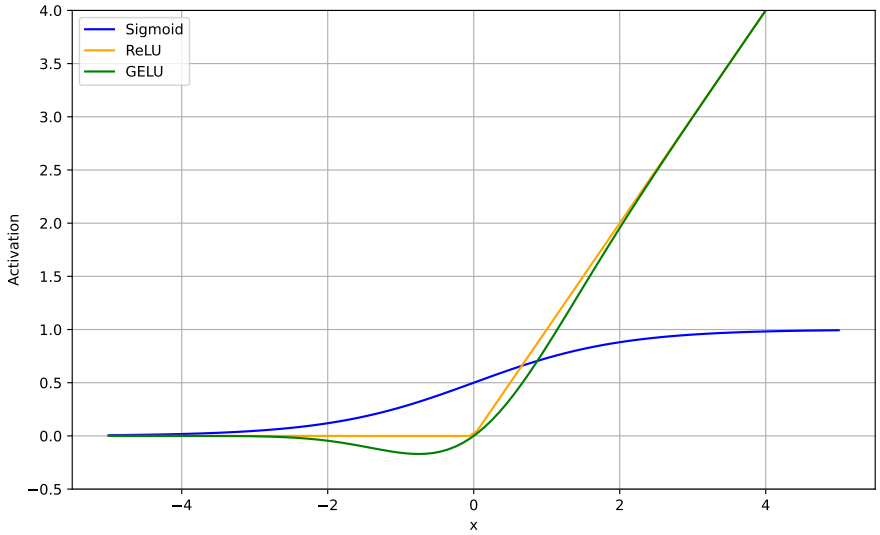


Figure 2.3: Graph for the sigmoid, ReLU and GELU activation functions.

update the weights of dense layers, due to the vanishing gradient problem (Chollet, 2018, p. 246). This is because large input values are squeezed into a small range, which subsequently results in a small derivative. As the derivative is used to calculate the gradient, the gradient becomes smaller and smaller with each layer it is calculated for. This means the bottom layers (the ones close to the input) of the network will receive a very weak error signal, hindering the learning process.

Rectified Linear Unit The rectified linear unit (ReLU) function is defined as:

$$\text{ReLU}(x) = \max(0, x) \tag{2.7}$$

ReLU is computationally efficient and helps mitigate the vanishing gradient problem (Ide and Kurita, 2017). This is because, during *backpropagation* (see Section 2.2.4), its derivative will always be ReLU'$(x) = 1$ for $x \geq 0$, solving the issue of having smaller and smaller gradients when calculating the loss for each consecutive layer, such as when using a sigmoid function. However, as ReLU'$(x) = 0$ for $x < 0$, this causes neurons receiving negative

inputs to not be updated during backpropagation, leading to the so-called "dying ReLU" problem. This problem brought about the development of many different variants of the ReLU function, with the aim of mitigating its effects, such as the Gaussian error linear unit (GELU) function (Stergiopoulos et al., 2022).

Gaussian Error Linear Unit The gaussian error linear unit (GELU) function is a smooth approximation of the ReLU function, which has been shown to improve the performance of neural networks in certain settings (Hendrycks and Gimpel, 2020). The formula for the GELU function is:

$$\text{GELU}(x) = 0.5x \left(1 + \tanh\left(\sqrt{2/\pi}(x + 0.044715x^3)\right)\right) \tag{2.8}$$

The GELU function combines the advantages of both the sigmoid and ReLU functions, providing a smooth and differentiable function that mitigates the vanishing gradient problem and the dying ReLU problem, along with being able to handle dropout regularization (Nguyen et al., 2021). One state-of-the-art application of this function is found in the BERT model (Devlin et al., 2019), short for Bidirectional Encoder Representations from Transformers, whose architecture is described in Section 2.3.

Softmax The softmax function is used to convert the output of a neural network into a probability distribution over the output categories it is trained to predict (also known as *classes*). It is defined as:

$$\text{softmax}(x_i) = \frac{e^{x_i}}{\sum_{j=1}^{N} e^{x_j}} \tag{2.9}$$

where x_i is the output of the i-th neuron, and N is the total number of neurons in the output layer.

2.2.2 Loss Functions

A *loss function* is used to measure the error of a model's predictions. The type of loss function used to calculate the error depends on the problem being approached. Binary classification usually involves the use of binary cross entropy (BCE), while the generalized cross entropy function can be used for multi-class classification settings. Regression tasks, on the other

hand, are usually approached using mean absolute error (MAE) or mean squared error (MSE). In this study, we use BCE for the classification tasks, and MAE and MSE for the regression tasks.

Cross Entropy Cross entropy is a loss function used to measure the difference between two probability distributions. For a single sample, it is defined as:

$$CE_{sample} = -\sum_{i=1}^{n} t_i \log(p_i) \tag{2.10}$$

where n is the number of classes (the entire vocabulary size), t_i is the true label of class c (1 for the correct class, 0 for all others), and p_i is the predicted softmax probability of the i-th class.

Binary Cross Entropy BCE is a special case of cross entropy, used for binary classification tasks, where the number of classes is $n = 2$. Its definition for a single training sample is:

$$\begin{aligned} BCE_{sample} &= -\sum_{i=1}^{2} t_i \log(p_i) \\ &= -t \log(p) - (1-t) \log(1-p) \end{aligned} \tag{2.11}$$

where t is the true label (1 for the positive class, 0 for the negative class), and p is the predicted probability of the positive class.

For N total samples, the BCE loss function is defined as:

$$BCE = -\frac{1}{N} \sum_{i=1}^{N} t_i \log(p_i) - (1-t_i) \log(1-p_i) \tag{2.12}$$

where t_i is the true label of the i-th sample, and p_i is the predicted probability of the i-th sample belonging to the positive class.

Mean Absolute Error MAE is a loss function used in regression tasks. It is defined as:

$$MAE = \frac{1}{N} \sum_{i=1}^{N} |y_{e,i} - y_{p,i}| \tag{2.13}$$

where $y_{e,i}$ is the expected output of the i-th sample, $y_{p,i}$ is the predicted output, and N is the total number of samples.

Mean Squared Error To obtain MSE, the absolute value in the MAE formula is replaced with the square of the difference:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_{e,i} - y_{p,i})^2 \tag{2.14}$$

2.2.3 Optimizer

An *optimizer* is an algorithm which is used to minimize the loss of a neural network's output by updating its weights during training. There are many types of optimization algorithms and the choice of optimizer depends on the problem being approached and the type of neural network being used.

Gradient descent is the most basic type of optimizer. It calculates the gradient of the loss function with respect to all the parameters in the network, which can make its calculation very expensive. The process by which gradient descent is implemented with regard to all parameters of a network can be written as:

$$\theta_{t+1} = \theta_t - \alpha\nabla_\theta E(\theta) \tag{2.15}$$

where $\theta = \{w_1, w_2, \ldots, w_n, b_1, b_2, \ldots, b_n\}$ is the set of weights and biases, α is the learning rate, and $E(\theta)$ is the loss function.

Variations of gradient descent have been developed to mitigate this issue, such as stochastic gradient descent (SGD), which calculates the gradient of the loss function with respect to a random sample of the training data, and mini-batch gradient descent, which calculates the gradient of the loss function with respect to a small batch of the training data.

Other optimization algorithms based on gradient descent have also been created. These make the training process more efficient, increasing the speed of convergence and reducing the risk of getting stuck in local minima. Two popular examples are AdaGrad (Adaptive Gradient Algorithm) (Lydia and Francis, 2019) and RMSProp (Root Mean Square Propagation),[7] which both

[7]RMSProp was never presented in a publication. It was only presented by Geoffrey Hinton in a course presentation: `http://www.cs.toronto.edu/~tijmen/csc321/slides/lecture_slides_lec6.pdf`

modify the learning rate of the parameters of a network. AdaGrad updates the learning rates by accumulating squared gradients from all previous time steps:[8]

$$\begin{aligned} g_t &= \nabla E(\theta_t) \\ G_t &= G_{t-1} + g_t \cdot g_t^T \\ \theta_{t+1} &= \theta_t - \frac{\alpha}{\sqrt{G_t + \epsilon}} \cdot g_t \end{aligned} \tag{2.16}$$

where g_t is the gradient of the loss function E with respect to the parameters at time step t, G_t is the sum of the squared gradients up to time step t, θ_t is the parameter at time step t, α is the learning rate, and ϵ is a small constant which prevents division by zero.

RMSprop uses an exponential moving average (EMA) of squared gradients, which diminishes the influence of past gradients:

$$\begin{aligned} E[g^2]_t &= \rho E[g^2]_{t-1} + (1-\rho) g_t^2 \\ \theta_{t+1} &= \theta_t - \frac{\alpha}{\sqrt{E[g^2]_t + \epsilon}} g_t \end{aligned} \tag{2.17}$$

where $E[g^2]_t$ is the EMA of the squared gradients at time step t and ρ is the decay rate. For $t = 0$, $E[g^2]_0$ is initialized as a zero vector.

Incorporating the advantages of AdaGrad and RMSProp, Adam (Adaptive Moment Estimation, Kingma and Ba (2017)) is an optimizer which has become rather popular in the past few years for deep learning tasks. Equation (2.18) shows the update rule for Adam:

$$\begin{aligned} m_t &= \beta_1 m_{t-1} + (1-\beta_1) g_t \\ v_t &= \beta_2 v_{t-1} + (1-\beta_2) g_t^2 \\ \hat{m}_t &= \frac{m_t}{1-\beta_1^t} \\ \hat{v}_t &= \frac{v_t}{1-\beta_2^t} \\ \theta_t &= \theta_{t-1} - \frac{\alpha}{\sqrt{\hat{v}_t} + \epsilon} \hat{m}_t \end{aligned} \tag{2.18}$$

where m_t is the first moment of the gradients at time step t (i.e., the EMA of the gradients), v_t is the second moment of the gradients at time step t (i.e.,

[8] In the context of training neural networks, a time step can be thought of as the cycle of feeding an input forward through a network and backpropagating the error.

the EMA of the squared gradients), β_1 and β_2 are, respectively, the decay rates for the first and second moments, and $\hat{m}_t$ and $\hat{v}_t$ are, respectively, the bias-corrected first and second moments.

The bias correction consists in dividing the first and second moments by $1-\beta_1^t$ and $1-\beta_2^t$, respectively, where $\beta_1 = 0.9$ and $\beta_2 = 0.999$ are the default values for the decay rates (Kingma and Ba, 2017). This is done because, otherwise, during the first training steps the first and second moments would make the learning rate α too high. As such, the bias correction helps mitigate this issue by making the term $\frac{\alpha}{\sqrt{\hat{v}_t}+\epsilon}\hat{m}_t$ smaller in Equation (2.18).

The first and second moments of the gradients modify the learning rate α of the parameters, which optimizes the speed of the training process.

2.2.4 Training

Neural networks learn through a process known as backpropagation (Chollet, 2018, p. 52). A neural network is trained to perform a certain task by making it predict an output based on input data and comparing said output to the expected output, also known as the *ground truth*. Compared to the forward pass used to obtain the output, backpropagation is carried out in inverse order. During backpropagation, the error ϵ is calculated as:

$$\epsilon = y_\text{e} - y_\text{p} \tag{2.19}$$

where y_e and y_p are the expected and the predicted output, respectively. The error is propagated backwards through the network by calculating the gradient ∇E of the loss (or error) function E with respect to the weight array $\mathbf{w}$. This means calculating the partial derivative $\frac{\partial E}{\partial w_i}$ of the loss function with respect to each weight $w_i \in \mathbf{w}$, with $i \in [0, n]$.

The purpose of doing this is that the gradient of the loss function essentially represents a vector which indicates the direction towards which the error decreases. Since the objective of the backpropagation algorithm is to minimize the loss, this is done by modifying weights and biases based on the gradient of the error. That is, we calculate what change in the weights and biases produces a decrease in the loss. To find the partial derivative of the error E with respect to the weights and the biases, we can use the *chain rule* of differentiation (Chollet, 2018, p. 51):

$$\frac{d}{dx}f(g(x)) = f'(g(x)) \cdot g'(x) \tag{2.20}$$

In the case of backpropagation, this rule is used to express the partial derivative of the loss function with respect to the weights and biases as a product of the derivatives of the intermediate variables. A loss function $E(g(z(\mathbf{w}, b; \mathbf{x})))$ with respect to a single perceptron is a function of the output $y = g(z(\mathbf{w}, b; \mathbf{x}))$, which is a function of the weighted sum $z(\mathbf{w}, b; \mathbf{x})$, which in turn is a function of the input $\mathbf{x}$, weights $\mathbf{w}$ and biases b. Therefore, we can use the chain rule to express the partial derivative $\frac{\partial E}{\partial w}$ with respect to a single weight w associated with a single perceptron as:

$$\begin{aligned} \frac{\partial E}{\partial w} &= \frac{\partial E}{\partial y} \cdot \frac{\partial y}{\partial w} \\ &= \frac{\partial E}{\partial y} \cdot \frac{\partial y}{\partial z} \cdot \frac{\partial z}{\partial w} \end{aligned} \tag{2.21}$$

where $y = g(z)$ is the output of the neuron after applying the activation function, $z = \mathbf{x} \cdot \mathbf{w} + b$ is the weighted sum of the inputs, and w is the weight.

With respect to the biases, the formula becomes:

$$\frac{\partial E}{\partial b} = \frac{\partial E}{\partial y} \cdot \frac{\partial y}{\partial z} \cdot \frac{\partial z}{\partial b} \tag{2.22}$$

where b is the single bias for one perceptron. These calculations need to be done for each weight and bias in the network, i.e., for every single trainable parameter.

The weights and biases are then adjusted in the direction that minimizes the error, using an optimization algorithm, such as gradient descent (see Section 2.2.3):

$$w_{\text{new}} = w_{\text{old}} - \alpha \frac{\partial E}{\partial w} \tag{2.23}$$

$$b_{\text{new}} = b_{\text{old}} - \alpha \frac{\partial E}{\partial b} \tag{2.24}$$

where α is the *learning rate*, a hyperparameter set to control the step size of the updates, i.e., by how much we are moving toward the minimum of the loss function. On one hand, if the learning rate is too small the training process will take a long time to converge and may end up in a shallow local minimum, rather than a lower local minimum. On the other hand, if the learning rate is too large, the training process may not converge at all and bring the weights and biases to essentially random values.

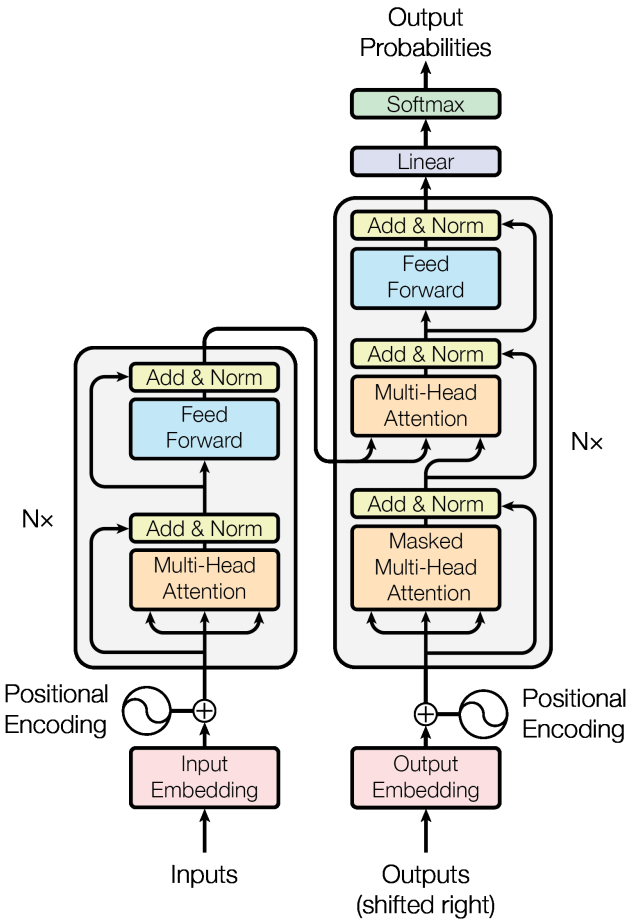


Figure 2.4: Transformer architecture. Image credit: Vaswani et al. (2017).

The backpropagation process is repeated until the error is deemed sufficiently small, or for a maximum number of preset iterations, commonly known as *epochs*.

2.3 Transformers

Transformers are a type of neural network architecture which was introduced by Vaswani et al. (2017). Their introduction led to the creation of powerful

models which have achieved state-of-the-art results in many NLP tasks, including machine translation (Raffel et al., 2020), question answering (Devlin et al., 2019), and text summarization (Lewis et al., 2019).

Figure 2.4 shows the Transformer architecture, comprising an encoder (the main block on the left) and a decoder (on the right), which work in parallel. Both the encoder and the decoder are made up of N layers running in parallel ($N = 6$ in Vaswani et al. (2017)). The encoder is fed an input sentence, while the decoder is fed the target sentence, shifted right by one position.

The encoder and decoder can be used by themselves, or together in an encoder-decoder architecture:

- **Encoder-only**: An encoder-only application is, for example, text classification. A very popular encoder-only model which can be used for this is BERT (Devlin et al., 2019), which we discuss in Section 2.4.
- **Decoder-only**: A decoder-only model can be used for tasks such as text generation, where the decoder is fed an initial prompt and generates the rest of the text. An example of this is GPT-2 (Radford et al., 2019).
- **Encoder-decoder**: An encoder-decoder model is used for sequence-to-sequence tasks, such as machine translation or question answering. An example of this is BART (Lewis et al., 2020).

2.3.1 Self-Attention

The Transformer architecture is based on the attention mechanism, which has become a fundamental component of many state-of-the-art models, especially in NLP. The purpose of the *self-attention* mechanism introduced by Vaswani et al. (2017) is to increase performance in sequence-to-sequence tasks, especially for translation. This new approach far outperforms past architectures for such applications, including recurrent neural networks (RNNs). In sequence-to-sequence tasks, an encoder is used to produce representations of an input sentence, while a decoder generates the output sentence. The limitation of RNNs, which the attention mechanism strives to solve, is the limited scope of the context which can be used to generate the output. In the case of RNNs, this shortcoming is due to the fact that they can only use

previous hidden states to generate the next one. When using attention, conversely, all words are assigned weights which indicate how much the decoder should focus on them, i.e., pay "attention" to them. This expands the context window of the decoder to the whole input sentence, which allows it to generate more accurate outputs. The term "self-attention" refers to the fact that the attention mechanism is applied to the input sentence itself, rather than to a different sentence.

2.3.2 Encoder

The encoder is made up of N identical layers, each of which is made up of two sub-layers: a multi-head self-attention layer and a linear network with two fully connected layers. The multi-head attention block helps the encoder pay attention to different parts of the input sentence, while the feed-forward network (FFN) processes and combines the information from the attention block. The purpose of the "Add & Norm" blocks placed after each module is to preserve the information from before the module. The "Add" part, also called a "residual connection", consists in adding the original vector to the output vector, which has the purpose of facilitating the backpropagation of the gradient (Xu et al., 2023). The "Norm" part refers to the fact that the output vector is normalized, i.e., its values are scaled between 0 and 1.

2.3.2.1 Embeddings

Before being fed into the encoder block, a sentence is first tokenized, mapping characters, subwords and whole words to unique IDs. For example, given the sample sentence "NLP is really fun!" and using BERT_{base}'s tokenizer,[9] the sentence would be tokenized as ["nl", "##p", "is", "really", "fun", "!"] and the tokens would be given the following input IDs: [17953, 2361, 2003, 2428, 4569, 999]. Other tokens may be added later on in order to process the input, for instance at the beginning or end of the sentence.

Each ID number is then converted into a word embedding E of fixed length d_{model} by passing it through an embedding layer:

$$E(ID) = [e_1, e_2, ..., e_{d_{model}}] \tag{2.25}$$

[9] `https://huggingface.co/bert-base-uncased`

where e_i is the i-th element of the embedding vector. Taking into account the sample sentence above and an embedding size of $d_{model} = 768$, the new encoded sentence would be represented by a 6×768 matrix of numbers.

In order to make the model aware of the position of each word in the sentence, positional embeddings are added to the word embeddings. This way, the Transformer is given information not just about the semantic content of the input tokens, but also about their position in the sentence. The positional embeddings P are calculated using sine and cosine functions:

$$P_{(pos,2i)} = \sin\left(\frac{pos}{10000^{2i/d_{model}}}\right) \tag{2.26}$$

$$P_{(pos,2i+1)} = \cos\left(\frac{pos}{10000^{2i/d_{model}}}\right) \tag{2.27}$$

where pos is the position of the token in the sequence, $0 \leq i \leq d_{model}$ is the index of the embedding, and d_{model} is the embedding size. Sine and cosine are used because they are defined for $-\infty < x < \infty$, with values between -1 and 1, which makes them similar in size to the values contained in the word embeddings. This means the meaning of the positional embeddings will not dominate the meaning of the word embeddings. However, sine and cosine with a high frequency (i.e., with a big coefficient to pos) would map every even and odd positions pos to the same embedding vectors, respectively. To avoid this, a $10,000$ factor is used in the denominator, which decreases the frequency of the functions and makes them return non-repeating values across a wide range for each embedding vector.

Given the same example sentence as before, the positional embedding of the token "is" ($pos = 2$) would be calculated as:

$$\begin{aligned} P_{(2,0)} &= \sin\left(\frac{2}{10000^{2\times 0/768}}\right) = 0.909 \\ P_{(2,1)} &= \cos\left(\frac{2}{10000^{2\times 1/768}}\right) = -0.373 \\ &\dots \\ P_{(2,767)} &= \cos\left(\frac{2}{10000^{2\times 767/768}}\right) = 1.0 \end{aligned} \tag{2.28}$$

with the full sentence's positional embeddings being illustrated in Table 2.1. These functions generate a unique embedding for each position, which is then added element-wise to the word embeddings.

pos	Token	*P*			
0	nl	0,0	0,1	...	0,767
1	##p	1,0	1,1	...	1,767
2	is	2,0	2,1	...	2,767
3	really	3,0	3,1	...	3,767
4	fun	4,0	4,1	...	4,767
5	!	5,0	5,1	...	5,767

Table 2.1: Positional embeddings for the sample sentence ($d_{model} = 768$).

The final input vector $x_i(T_i) \in X = [x_1, x_2, ..., x_n]$ of a token $T_i \in T = [T_1, T_2, ..., T_n]$, with T being the full tokenized sentence and X the final input matrix, is the element-wise sum of its word embeddings and the positional embeddings:

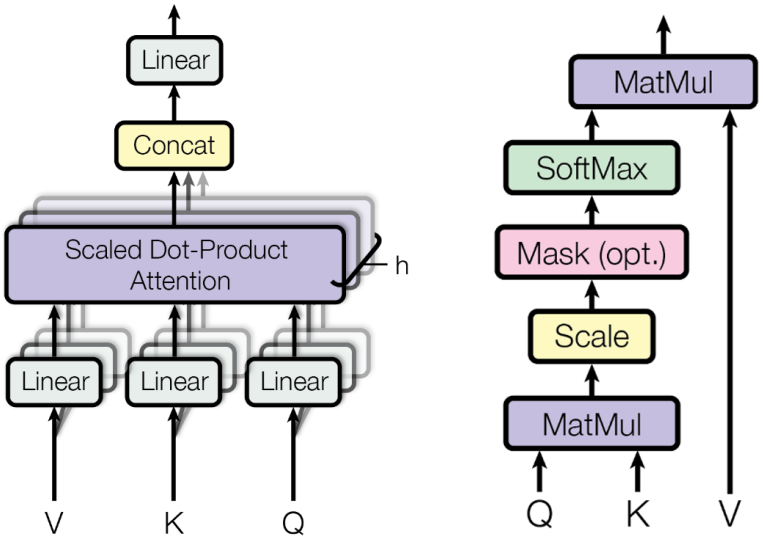
$$\begin{aligned} x_i(T_i) &= E_i(T_i) + P_i(T_i) \\ &= [e_{i,1}, e_{i,2}, ..., e_{i,d_{model}}] + [p_{i,1}, p_{i,2}, ..., p_{i,d_{model}}] \end{aligned} \tag{2.29}$$

where $E_i(T_i)$ is the word embedding of the token T_i, $P_i(T_i)$ is its positional embedding, and d_{model} is the embedding size.

2.3.2.2 Multi-Head Attention

The purpose of the multi-head attention module is to allow the model to focus on different parts of the input sequence. This is done by using the concept of self-attention, introduced in Section 2.3.1.

The module is called "multi-head" because it is made up of multiple parallel attention layers ($h = 8$ in Vaswani et al. (2017)). The structure of each layer is laid out in Figure 2.5a, with the scaled dot-product attention sub-module being shown in Figure 2.5b. Q stands for "query", K for "key",



(a) Multi-head attention. (b) Scaled dot-product attention.

Figure 2.5: Multi-head attention module and scaled dot-product attention sub-module. Image credit: Vaswani et al. (2017).

and V for "value". The Q, K, and V matrices are calculated by multiplying the input matrix X by three different weight matrices, W^Q, W^K, and W^V. This calculation is carried out in the three sets of linear layers shown in Figure 2.5a.

The three resulting matrices are then fed into the dot-product attention, whose output is calculated as:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{2.30}$$

where d_k is the dimension of the query and key vectors. The dot product between Q and K^T represents the "MatMul" function shown in Figure 2.5b, while the "Scale" function is the division by $\sqrt{d_k}$. After applying the softmax function, the output is multiplied by the value vector ("MatMul").

2.3.3 Decoder

Just like the encoder, the decoder is also made up of N layers with different modules. The modules, also stacked in layers, are very similar to the en-

coder's, but with a few differences: the input is shifted to the right, and the scaled dot-product attention sub-module includes a mask which prevents the decoder from attending to future tokens.

2.3.3.1 Decoder Input

In this case, the output embeddings are shifted by one position, compared to the input embeddings. That is, given for example a question answering task, during training an input sentence might be made up of the tokens ["Are", "we", "done", "yet", "?", "<PAD>", "<PAD>"], while the output would be ["<BOS>", "Just", "a", "few", "chapters", "left", "."] (Tunstall et al., 2022). Conversely, during inference, the decoder would initially be fed only the beginning-of-sequence <BOS> token on step 0, and would then generate the next (first) token of the output sequence. Its output would then be added back into the input of the decoder (making it *auto-regressive*), with the new input being ["<BOS>", "Just"] for step 1, ["<BOS>", "Just", "a"] for step 2, and so on. This process is repeated until the ending-of-sequence <EOS> token is generated, or the maximum length of the output sequence is reached.

The <BOS> token is a special token that indicates the beginning of a sentence, while <PAD> is a padding token used to make all sentences in a batch have the same length. The decoder necessarily needs an output, even at step 0, when it has not yet produced an output token, which is why the <BOS> token is used.

2.3.3.2 Masked Multi-Head Attention

The first multi-head attention stack of layers is almost identical to the one found in the encoder, except for the fact that a mask is applied to the output of the scaled dot-product attention sub-module, in order to prevent the decoder from "looking ahead" during training (see Figure 2.5b). Given the MatMul matrix resulting from the dot product between the query and key vectors of the target sequence, a mask matrix containing 0s in the lower triangular part and $-\infty$ in the upper triangular part is applied by summing it to the MatMul matrix. The result of this operation is visualized in Table 2.2. When the softmax function is then applied, the values of the upper triangular part of the matrix are assigned a probability of 0.

Feeding the decoder the correct target sequence and applying a mask which ensures the model cannot "look ahead" can also be seen as an im-

	<BOS>	Just	a	few	chapters	left	.
<BOS>	0.7	$-\infty$	$-\infty$	$-\infty$	$-\infty$	$-\infty$	$-\infty$
Just	0.7	0.6	$-\infty$	$-\infty$	$-\infty$	$-\infty$	$-\infty$
a	0.7	0.6	0.5	$-\infty$	$-\infty$	$-\infty$	$-\infty$
few	0.7	0.6	0.5	0.4	$-\infty$	$-\infty$	$-\infty$
chapters	0.7	0.6	0.5	0.4	0.3	$-\infty$	$-\infty$
left	0.7	0.6	0.5	0.4	0.3	0.2	$-\infty$
.	0.7	0.6	0.5	0.4	0.3	0.2	0.1

Table 2.2: Mask applied to the MatMul matrix resulting from the dot product between the query and key vectors.

plementation of "teacher forcing" (Williams and Zipser, 1989), an approach whereby the model is fed "the previous ground truth labels and not the current or future ones" (Tunstall et al., 2022). Teacher forcing is used so that the model bases itself on the ground truth for each step, rather than on its own predictions, which are likely to be inaccurate at the beginning of training.

2.3.3.3 Decoder Multi-Head Attention

This module is identical to the multi-head attention module found in the encoder, but in this case the key and query vectors are the outputs of the encoder, while the values are the outputs of the masked multi-head attention module. Just like in the encoder, the output of the multi-head attention is then added to the residual connection and normalized, and then passed through an FFN, before being added to the residual connection and normalized again.

2.3.3.4 Decoder Output

The output of the decoder is finally passed through a linear layer and then through a softmax function, which returns a probability distribution over the vocabulary. There are multiple ways to choose the token to be used as output. The simplest way is the "greedy search" strategy, which entails choosing the token with the highest probability. Another strategy is "beam search", which entails keeping track of the k most probable tokens at each step, where k is the number of "beams" or partial hypotheses. The following beams are then selected based on all the possible next token continuations, repeating the process until the <EOS> token is generated (Tunstall et al., 2022).

2.4 BERT

BERT ("Bidirectional Encoder Representations from Transformers") is a Transformer-based language representation model (Devlin et al., 2019). Since BERT is based on the encoder of the Transformer architecture, it can attend to all tokens at once, which makes it "bidirectional". The application of the model entails two main phases: pre-training and fine-tuning. Pre-training is carried out on a variety of unsupervised tasks, while fine-tuning is done with labeled data after loading the pre-trained weights. Fine-tuning means training the model on a certain target task, also referred to as a "downstream" task, and is done by implementing minor changes with relation to the architecture of the pre-trained model.

The base version introduced in Devlin et al. (2019), BERT_{base}, has $L = 12$ layers of encoders, $A = 12$ layers for the attention heads, and $H = 768$ hidden units. A larger version, BERT_{large}, has $L = 24$ layers of encoders, $A = 16$ layers for the attention heads, and $H = 1024$ hidden units.

2.4.1 Input Representation

The input representation used for BERT is able to represent unambiguously both a single sentence and a pair of sentences. Each input is referred to as a "sequence", which is always started by a [CLS] classification token, henceforth referred to as "C". In sequences containing two sentences, the sentences are separated by a [SEP] token which lets the model know where

the first one ends and the second begins. The [SEP] token is also placed at the end of the sequence to mark its ending. Each token is turned into an embedding, just like in the Transformer architecture, which are summed to positional embeddings. Additional BERT segment embeddings (also known as "token type" embeddings) are added to the representations to indicate whether a token belongs to the first or second sentence (if the task at hand requires such a distinction to be made).

2.4.2 Pre-Training

BERT is pre-trained on two unsupervised tasks: masked language modeling (MLM) and next sentence prediction (NSP).

2.4.2.1 Masked Language Modeling

Language models are usually trained to predict the next token based solely on the preceding (left-to-right) or following (right-to-left) tokens. This is only possible when the model is unidirectional and can only attend to the left or right context. As the Transformer encoder attention gives the model access to all tokens at once, this approach is not possible, as the model would have access the same token it is trying to predict. During training, using this approach would mean that the model would learn to simply copy the token, instead of learning its relationship with the surrounding context.

In order to train BERT to predict tokens in a sentence, 15% of the tokens in each sequence are picked at random for masking. Given a sequence of n tokens, and a token T_i picked for masking, with $i \in \{1, \ldots, n\}$, when the i-th token is picked, it is replaced by a [MASK] token 80% of the time, by a random token 10% of the time, or by the original token 10% of the time. This is done because simply replacing the token every time it is picked would cause a mismatch between the pre-training and fine-tuning, since the [MASK] token is not going to be present in the fine-tuning data. The model is then trained to predict the original token T_i based on the context provided by the other tokens in the sequence, using cross-entropy loss.

2.4.2.2 Next Sentence Prediction

Tasks such as question answering (QA) and natural language inference (NLI) depend heavily on a model learning the relationship between two entire sen-

tences. BERT does not learn these relationships during the MLM task, as the model is only trained to predict individual tokens. In order to learn these relationships, the model is also trained on the next sentence prediction (NSP) task.

An unsupervised dataset is built automatically from a corpus. A sentence is picked and 50% of the time it is paired with the next sentence in the corpus, while the other 50% of the time it is paired up with a random one. The sequence is assigned a binary label indicating whether the second sentence follows the first ($label = 1$), or not ($label = 0$). The C classification token, which contains a representation of the whole sentence, is used to predict the label using cross-entropy loss.

According to the authors, this task greatly improves BERT's performance on QA and NLI tasks, despite being a simple binary classification task.

2.4.3 Fine-Tuning

Fine-tuning BERT for a specific downstream task simply involves feeding the model the required inputs and applying the desired output layer on top of the pre-trained model. As far as the input is concerned, for tasks involving a single sentence (e.g, classification tasks), no [SEP] token is used. In tasks which require a pair of sentences (e.g., QA and NLI), the sentences are fed to the model separated by the [SEP] token. As regards the output, the representations of the tokens belonging to the sentences can be used for token-level tasks (e.g., QA or sequence tagging), while the C token, which contains the aggregate representation of the sequence, can be used for sentence-level tasks (e.g., classification). Given the output vector V of the encoder's last FFN layer, with elements $v_i \in V$ where $V \in \mathbb{R}^{768}$, the C token hidden state is v_0. The tokens belonging to the first sentence are $v_1, \ldots, v_n$, and the tokens belonging to the second sentence are $v_{n+2}, \ldots, v_{n+m}$, where n is the number of tokens in the first sentence and m is the number of tokens in the second sentence. The second sentence starts at v_{n+2} rather than v_{n+1} because v_{n+1} is the [SEP] token.

2.4.4 BERT for Sequence Classification

As illustrated in Figure 2.6, sequence classification with BERT involves feeding the C token hidden state (a 768-dimensional vector) to a FFN comprising

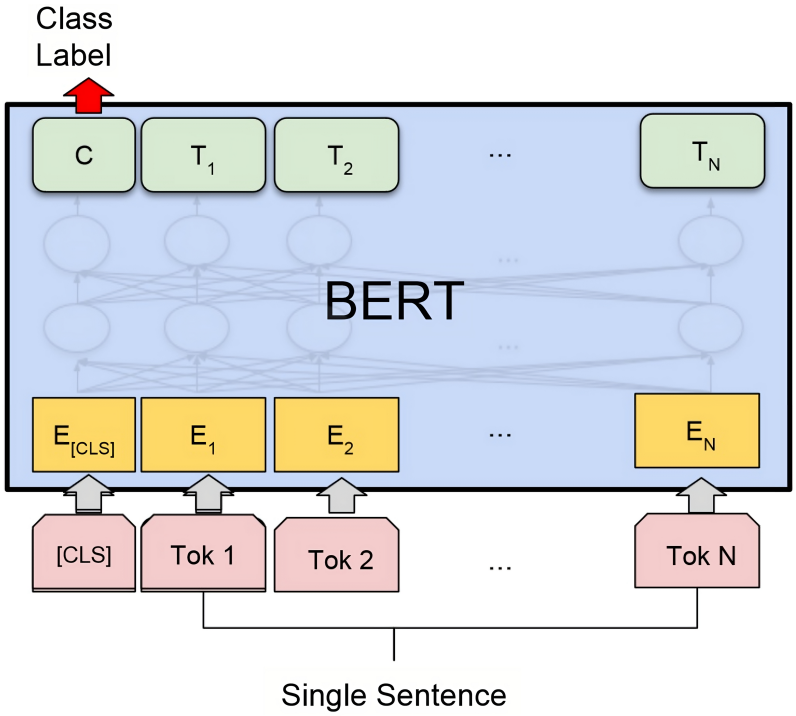


Figure 2.6: BERT for classification of a single sentence. Image credit: Devlin et al. (2019).

a first layer with the same dimensionality as the hidden state and an output layer whose dimensionality can vary depending on how many classes are being predicted.

For binary classification, where $N = 2$ classes $c \in \{0, 1\}$ are predicted, a 1D output layer with sigmoid activation function can be used as the output of the linear layer. In this case, given the output value y, the class c for a sample can be set to $c = 0$ if $y \geq 0.5$, or $c = 0$ otherwise. A softmax activation function can also be used, connecting the linear layer to an output layer of dimension $N = 2$, and assigning $c = argmax(\mathbf{y})$, where $\mathbf{y} = [y_0, y_1]$ is the output vector of the softmax function.

For multi-class classification, where $N > 2$ classes $c \in \{1, \ldots, N\}$ are predicted, a softmax activation function can be used, connecting the linear layer to an output layer of dimension N, assigning $c = argmax(\mathbf{y})$, where $\mathbf{y} = [y_0, \ldots, y_N]$ is the output vector of the softmax function.

In our study, we use BERT and other similar Transformer models (e.g.,

RoBERTa (Liu et al., 2019)) with softmax output layers for our classification tasks (discussed in Sections 5.3 and 5.4).

2.4.5 BERT for Regression

Training BERT for regression tasks is similar to training it for classification tasks, with the difference that the output layer is a 1D layer with a linear activation function. The output of the linear layer is the predicted value of the regression task.

In order to conduct our experiments involving regression tasks, discussed in Section 5.5, we apply this output configuration to BERT and mBERT (Devlin et al., 2019) models.

2.5 Evaluation Metrics

The performance of a model is evaluated based on metrics which depend on the problem being approached. This section presents the metrics used to evaluate classification and regression tasks.

2.5.1 Classification

A classification task involves a model classifying an input instance by predicting its class label. The number of classes which can be associated to the input instance is an integer number $N \geq 2$. For binary classification ($N = 2$), commonly used metrics are: precision, recall and F_1-measure of the positive outcomes. The instance label is positive if the instance possesses a certain characteristic; otherwise, it is negative. Precision and recall measure a model's performance in predicting positive class labels, and are needed to calculate the F_1-measure.

The distribution of the classifications produced by a model can be visualized effectively in a confusion matrix, such as the one displayed in Figure 2.7, which shows an example with a total amount of 1,000 classified instances. The first and second columns of the matrix represent, respectively, the actual number of positive (405) and negative (595) instances. Conversely, the first and second rows represent, respectively, the number of instances predicted as positive (415) and negative (585) by the model. The diagonal of

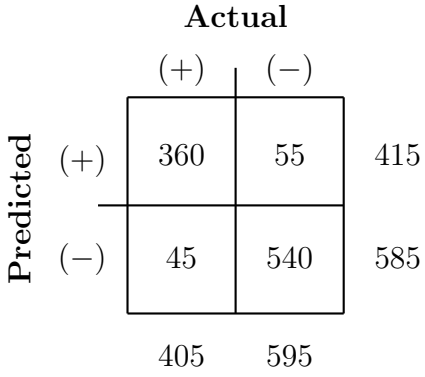


Figure 2.7: An example confusion matrix.

the matrix (360 and 540) represents the correctly classified instances, known respectively as the true positives (TP) and the true negatives (TN). The off-diagonal elements (55 and 45) represent the incorrectly classified instances, known respectively as the false positives (FP) and the false negatives (FN).

The precision P of a model is the ratio of the correct positive predictions to the total number of positive predictions:

$$P = \frac{TP}{TP + FP} \tag{2.31}$$

The recall R is the ratio between the correct positive predictions and all possible positive predictions:

$$R = \frac{TP}{TP + FN} \tag{2.32}$$

In this study, classification tasks are all approached in binary settings. For binary classification, the F_1-measure is evaluated solely on the positive label and is calculated as the harmonic mean of precision and recall:

$$F_1 = 2\frac{P \cdot R}{P + R} \tag{2.33}$$

In the example shown in Figure 2.7, the precision would be $P = \frac{360}{360+55} = 0.867$, the recall would be $R = \frac{360}{360+45} = 0.889$, while the F_1-measure would be $F_1 = 2\frac{0.867 \cdot 0.889}{0.867+0.889} = 0.878$

2.5.2 Regression

In regression problems, a model is tasked to predict a scalar output. In this thesis, the output of a model is evaluated for regression tasks by using mean absolute error (MAE) and mean squared error (MSE). The equations for MAE and MSE, which we use for the experiments discussed in Section 5.5, are presented in Section 2.2.2.

Chapter 3

Related Work

This chapter presents related research on the topic of hate speech, with particular attention to the domain of inceldom. Section 3.1 showcases a variety of studies conducted on the topic of discourse produced by incels. Section 3.2 provides an overview of past research on the topic of hate speech identification. Section 3.3 presents the most relevant studies on the topic of hate speech forecasting.

3.1 Incel Discourse

From a sociological point of view, the phenomenon of inceldom, while recent, has already been the subject of a variety of studies. One of the most comprehensive summarizations of the topic up to 2017 was produced by Nagle (2017), who provides an overview on incels from the standpoint of U.S. politics, the alt-right, and feminism, among others.

In the field of sociolinguistics, most studies on inceldom have focused on the linguistic properties of incel corpora, predominantly adopting qualitative approaches. For example, Tranchese and Sugiura (2021) compare incel discourse from Reddit forums to the language used in pornography and highlight its misogynistic implications.

Papadamou et al. (2020) conduct a cross-platform study on incel profiling, by collecting $6.5k$ YouTube videos shared by users in Incel forums within Reddit, while also examining the YouTube recommendation algorithm. Their findings show that incel activity on YouTube is increasing, stirring towards the dissemination of incel views.

Past studies have also relied on the Pushshift Reddit API to build corpora within the linguistic domain of inceldom.[1] For example, Farrell et al. (2020) study the language of inceldom in seven subreddits, analyzing various specific subcategories of misogynous speech. Gothard (2020) also uses Reddit to study incel language, but with a more qualitative approach, studying the specific features that make incel speech a highly opaque and characteristic language. Such features include exclusive jargon such as "foids", "fakecel", "roastie", and "jfl" (acronym for "just fucking lol").

3.2 Hate Speech Identification

This section provides an overview of related work on hate speech identification, focusing specifically on research which can be considered relevant with relation to inceldom, taking into account monolingual and multilingual approaches. A large share of past research on this topic involves leveraging supervised datasets compiled specifically for the downstream task of hate speech identification.[2]

English Among English-language datasets, one of the most cited is probably the one released by Davidson et al. (2017), who compile tweets annotated with multi-class labels ("hate speech", "offensive", "neither"). They approach the task of identifying hate speech with a variety of machine learning models, finding that logistic regression and support vector machines (SVM) achieve the best performance.

De Gibert et al. (2018) release a dataset containing posts extracted from `stormfront.org`, a white supremacist forum. The dataset is annotated for hate speech using a custom annotation tool, and comes with various baselines for other researchers to use as a basis of comparison. The authors also conduct a qualitative analysis and provide majority, SVM, CNN and LSTM baselines for the hate speech identification task.

The English dataset released for Task 5 of SemEval 2019 (Basile et al., 2019), also known as "HatEval", contains $13k$ tweets which were annotated with binary labels for hate speech against migrants and women. The task

[1] The two largest incel subreddits to ever be hosted on Reddit, /r/incels and /r/braincels, were respectively banned in 2017 and 2018.

[2] An exhaustive list can be found at: `https://github.com/leondz/hatespeechdata`

has two sub-tasks, for which secondary labels are provided, with a first set of labels indicating whether the target is a group or an individual, and a second indicating whether the hateful speech was aggressive or not. The best performing models used by the teams included SVMs, CNNs and LSTMs.

Jaki et al. (2019) adopt a mixed approach, mainly focusing on text profiling, with their discourse analysis suggesting that incel language is not as coherent as previously assumed. They also employ a multichannel CNN, using $50k$ Incels.me messages, $50k$ neutral texts composed of $40k$ paragraphs from random English Wikipedia articles, and $10k$ random English tweets.

Caselli et al. (2021) retrain BERT_{base}[3] in an unsupervised way on the MLM task using the RAL-E dataset, a dataset built from messages extracted from subreddits (Reddit sections) which were banned because of the hateful, offensive or abusive nature of their content. Doing so, they obtain a model which they call *HateBERT*, capable of outperforming BERT_{base} on the downstream task of hate speech identification on various benchmark datasets. We take inspiration from this approach and apply it to our novel datasets, finding similar results (see Sections 5.3 and 5.4).

Mathew et al. (2021) build a dataset called *HateXplain* from Twitter and Gab posts, annotated with a multi-class label based on whether the post is "offensive", expesses "hate", or is "normal". The annotation also contains labels for the target of hate and rationale arrays. This last element in particular represents the particularity of the dataset and helps improve the performance and explain the predictions of the model, since the arrays provide information on which tokens human annotators considered important, thus leading to their annotation decision. The authors also release a BERT model fine-tuned on the dataset.

Mollas et al. (2022) use a combination of YouTube and Reddit comments to build a dataset with binary and multi-class labels. They provide baselines by using various machine learning algorithms and deep learning models, e.g., Naive Bayes, SVMs, CNNs, long short-term memory (LSTM) networks, and Transformer-based models, such as BERT.

Spanish For Spanish, the dataset compiled for IberEval 2018's automatic misogyny identification shared task (Fersini et al., 2018) provides a set of tweets annotated with binary labels for misogyny. The annotations also include categorizations for the type of misogyny and information on whether

[3] `https://huggingface.co/bert-base-uncased`

the target was an individual or women at large. Participating teams approached the classification task with a variety of approaches, most of them using ensembles of classifiers (EoC) and SVMs.

The 2019 Spanish HatEval dataset (Basile et al., 2019) contains $6.6k$ tweets and, just like its English counterpart, is annotated with binary labels for hate speech against immigrants and women. The tasks are the same as those approached with the English dataset. In this case, the best performing teams adopted a variety of approaches, e.g., SVMs, BERT models, and logistic regression.

Italian Datasets annotated for hate speech also exist in Italian, a few of which were prepared within the framework of EVALITA,[4] an evaluation campaign organized by the Italian Association for Computational Linguistics.[5] The 2018 edition of the campaign included a hate speech detection task (Bosco et al., 2018) for which two datasets were released, one built from Facebook posts and one from tweets. The Facebook dataset is annotated with an ordinal hate intensity label ("no hate", "weak hate", "strong hate") and a multi-class label indicating seven possible themes. The Twitter dataset contains a binary label for hate speech, an ordinal hate intensity rating (1-4), an ordinal aggressiveness rating ("no", "weak", "strong"), an ordinal offensiveness rating ("no", "weak", "strong"), and a binary irony label.

The 2020 edition of EVALITA included an automatic misogyny identification task (Basile et al., 2020), for which a $6k$-tweet dataset was compiled. The dataset is annotated with a binary label for misogyny and a binary label for aggressiveness. The best constrained runs (i.e., only using provided training data and lexicon) used a CNN, while the best unconstrained run (in which additional training data was allowed) used a BERT model. Muti and Barrón-Cedeño (2020) use AlBERTo (Polignano et al., 2019), a BERT model trained on Italian-language tweets, to approach the shared task, achieving top performance among all participants. The runners-up also experiment with BERT-based architectures, but approach the task using an ensemble technique (Lees et al., 2020).

[4] `https://www.evalita.it/`
[5] `https://www.ai-lc.it/en/`

Multilingual Research on this front has also been conducted in multilingual and cross-lingual settings,[6] with a number of multilingual datasets annotated for hate speech being used for this purpose.

A general purpose hate speech dataset in English, German and Hindi was released for the HASOC (Hate Speech and Offensive Content) track of the 2019 edition of the Forum for Information Retrieval Evaluation (FIRE) (Modha et al., 2019). The dataset includes $4.7k$ tweets and Facebook posts and is annotated for three sub-tasks. Sub-task A uses "hate", "offensive", or "neither" labels, while sub-task B uses "hate", "offensive", or "profane" labels. Sub-task C, a branch of sub-task A only entailing English and Hindi, contains labels providing information on whether the expressed hate speech is targeted or untargeted (i.e., addressing an entire group of people rather than a specific individual). The best results for sub-task A were obtained using an LSTM, while the best results for sub-tasks B and C were obtained using a BERT model, which at the time of the competition was still rather new.

Aluru et al. (2020) conduct an extensive multilingual study using 16 existing datasets in 9 different languages. They approach a monolingual scenario and a cross-lingual scenario. In the monolingual scenario, they find that for low-resource setups using LASER embeddings with logistic regression obtains the best results, while BERT-based models perform best when there is no scarcity of training data. In the cross-lingual scenario, they use training data from all the languages and test on a single language. Their results show that including training data from languages other than the target language improves the performance of the model, especially in few-shot and zero-shot settings, i.e., when there is little or no training data available for the target language. The cross-lingual results are especially good for Italian and Portuguese. In our study we find similar results, with English and Spanish aiding models with zero-shot predictions on Italian (see Section 5.3).

Pelicon et al. (2021) use a multilingual combination of datasets annotated for hate speech to improve the performance of Transformer models in zero-shot, few-shot and well-resourced settings. The few-shot settings are set up by increasing downstream training data in increments of 10%.

[6]With "multilingual", we refer to contexts in which two or more languages are involved, while by "cross-lingual" we mean scenarios in which a model is fine-tuned on one or more languages and then used to predict on another language and/or domain which was not used during training. The latter could also be referred to as "zero-shot", based on the specific experimental setup.

The authors find that the cross-lingual approach increases the performance of mBERT (Devlin et al., 2019) and cseBERT (Ulčar and Robnik-Šikonja, 2020), especially when merging datasets with languages that are linguistically similar to the language of the target task. We take inspiration from their dataset-merging approach and integrate it in our experiments with various MLM-enhnced models for hate speech detection in English and Italian.

Toraman et al. (2022) release a Twitter-based dataset of $200k$ tweets annotated for abusive language and hate speech in five domains, half in English and half in Turkish. They find that Transformer models outperform other models and that most of the performance is retained even when using just 20% of the collected training data. They also show that in both languages gender and religion are the domains which generalize best to other domains.

Gokhale et al. (2022) use MLM pre-training to improve the hate speech detection performance of BERT in Hindi and Marathi, separately. As training material, they use combinations of hateful and non-hateful content from a collection of $40M$ tweets. They find that using hateful tweets as pre-training material does not yield the best results, and that the results obtained using non-hateful content are similar or even better. They release the two pre-trained models,[7] one for Hindi and one for Marathi, along with two supervised benchmark datasets, one for each language, each containing $2k$ tweets annotated for hate speech. This approach is similar to the one used by Caselli et al. (2021) and the one we employ in this study. However, ours is different because we also mix two languages, English and Italian, for MLM pre-training.

As we have shown, there is no scarcity of general-purpose datasets annotated for hate speech. However, such resources are not necessarily applicable to the use-case of this study, either due to the source of the data only being partially compatible with the linguistic domain presently tackled (Pelzer et al., 2021; Pelicon et al., 2021) or because of the criteria according to which it was annotated (Zhou et al., 2022). Corpora built from incel platforms are rare: to our knowledge, the only study building a dataset using posts collected from incel forums was conducted by Pelzer et al. (2021), who do not make their dataset publicly available. Supported by these considerations and the study we conduct in Chapter 4, we therefore build new datasets from scratch to conduct the research presented in Chapter 5.

[7] `https://github.com/l3cube-pune/MarathiNLP`

3.3 Hate Speech Forecasting

Recently, more hate speech studies turn towards a new approach: *forecasting* the spread of hateful content within a sequence of posts.

Almerekhi et al. (2020) propose a model for toxicity triggering prediction by integrating text-based features as well as features related to discussion context and shifts in sentiment and topic flow. They show that non-toxic posts receiving toxic replies, which they refer to as *toxicity triggers*, contain detectable text features. Such features, when combined with the features indicating sentiment and topic flow variations, can be used to predict toxicity triggers.

Dahiya et al. (2021) compile a dataset of $4.5k$ tweets and their reply threads. They assign a hate score to chunks of threads by classifying them at the post level with a hate speech classifier and combining its labels with a hate score assigned to each post based solely on a model-independent lexicon. They find that longitudinal patterns of hate intensity among reply threads are diverse, with no significant correlation with the source tweet. As their approach involves labeling chunks of threads, and not whole threads at once, their approach differs from ours and is not directly comparable.

Lin et al. (2021) propose *HEAR*, a model which uses a post's semantic content, time features, and propagation structure to forecast the propagation of hateful content through thread replies. Their model, which was the first to use propagation features to forecast hate speech, outperformed several baseline models, including recursive neural networks (RvNN), cascade-LSTMs, and CNN-RNNs.

Meng et al. (2023) predict the intensity of hate that a tweet might carry through its reply chain by exploiting tweet threads and their semantic and propagating structures. This approach allows them to capitalize on the contextual information contained in a Twitter thread. Using three publicly available datasets, they show that their model, which they call *DRAGNET++*, outperforms six other baseline models, including *DRAGNET* (Sahnan et al., 2021). The model uses graph neural networks (GNN) to learn the semantic and propagation features contained in threads.

Since the models used by Lin et al. (2021) and Meng et al. (2023) use time features, they cannot be compared directly to our study, as our approach involves forecasting hate speech solely based on the textual features of the first post of a thread.

Chapter 4

The Language of Inceldom

This chapter discusses the peculiarities of incel language in the context of typical Internet forums and the implications of its lexical features. Section 4.1 introduces a brief study of the way incel lexicon changes over time. Section 4.2 presents unsupervised datasets created from forums frequented by incels. Section 4.3 reports the results of the diachronic study. Section 4.4 summarizes the contributions provided in the chapter.

4.1 Lexicon Diachronic Study

In order to study the phenomenon of inceldom from a primary language source, we study the lexicon features of messages posted on two Internet forums frequented by incels: *Incels.is* and *Il forum dei brutti*. The former is currently the most active incel platform in the world,[1] while the second is the biggest Italian-language incel forum.

One advantage of using these niche websites as opposed to communities hosted within massive social media platforms, such as Reddit, is that, as already mentioned in Section 2.1.4, these environments are more secluded and allow users more freedom, since moderation is more lax.[2] This means

[1]The /r/incels and /r/braincels subreddits, the most popular to date, were shut in 2017 and 2018, respectively.

[2]The predecessor to *Incels.is*, *Incels.me*, did get suspended by its domain provider in 2018 (`https://domain.me/the-suspension-of-incels-me/`). However, this only happened after a fatal terror attack perpetrated by members of the forum, in which 11 people were killed, which prompted Domain.me to shut the website down.

that the absence of third-party pressure to moderate content allows users to express their language freely, which is beneficial to corpus linguistic studies, since the language used is more genuine, less filtered, and in some cases much more extreme. This is very much the case with regard to *Incels.is*, where misogynous and racist sentiments spread rampant, as shown by the results of this study (see Section 5.5.1). That said, although this can represent an interesting feature, it can also have repercussions on the generalizability of the results, since the language used in these spaces might not be representative of the language used in more mainstream platforms. Therefore, further research could be conducted to verify whether compatible results can also be obtained when using data from social media platforms with broader audiences.

With the aim of addressing our first research question, which involves verifying whether we can build resources for the monolingual and cross-lingual identification of hate speech, misogyny, and racism, we first ensure the need to compile the data. Thus, we shed light on the way the language of inceldom evolves by studying the change in keyness (Kilgarriff, 2009) of specific sets of words, showing how the lexical features of these two communities change rapidly over time. We do this by conducting a "modern diachronic" (Partington, 2010) study of the use of key incel lexicon on the contents of the two forums. We calculate the keyness by using enTenTen20[3] as the reference corpus for English and itTenTen20[4] for Italian (Jakubíček et al., 2013).

By crawling thread URLs and scraping their contents, we create dumps of the two forums organized as rows of posts, each row containing the following content and metadata:

- Thread post number: the position of the post within the thread, the OP always being number 1.
- Username: the nickname used by the author of the post.
- Post content: as shown in Figure 4.1, the contents of a post are made up by the body of the message posted by the user and the message(s) they are quoting. The quoted content is shown inside the box with a green side accent, which also reports the username of the author being quoted on its top-left corner. A post can quote one or more preceding posts, fully or in part.
- Title: the title of the thread.

[3] `https://www.sketchengine.eu/ententen-english-corpus/`
[4] `https://www.sketchengine.eu/ittenten-italian-corpus/`

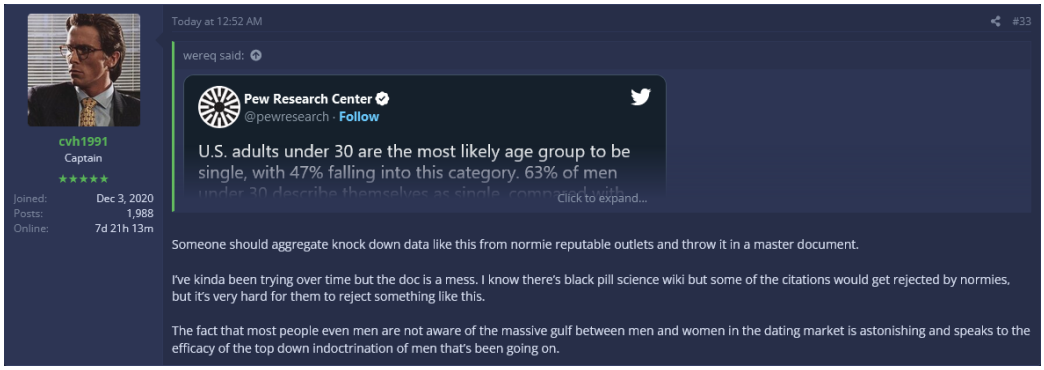


Figure 4.1: Screenshot of a post from the *Incels.is* forum.

- Thread ID: the numerical ID of the thread, unique for each thread across the whole forum.
- Post ID: the numerical ID of the post, unique for each post across the whole forum.
- Timestamp: the UTC time and date on which the message was posted.

In this study, we only consider the body of a post, disregarding quoted text; otherwise, words would be counted twice, once for the body and once for the quoted text, leading to misleading frequency values. It is also important to store the two sections separately, so that language models can be applied on the actual text produced by the user, rather than what they are replying to.

As regards *Incels.is*, in order to compile a list of characteristic incel lexicon, the keyness of lexical items was calculated across the entirety of the forum, up to October 2022. Preliminary candidates were selected by collecting single- and multi-word items that ranked in the top 500 for keyness, for a total of $1k$ analyzed items. Among these, only terms considered to be typical of incel language were examined. As mentioned in Section 2.1.3, racism and misogyny are very characteristic elements of the language of incels. As such, a simple way to choose characteristic terms for this speech community is manually evaluating racist and misogynous terms (or terms that are frequently associated to racist and misogynous contexts) and selecting those which are not typically found in general language, i.e., having high keyness scores. The evaluation of the individual terms was carried out by manually analyzing concordance lines in the corpus with the objective of verifying whether their use could be construed as being hateful. Although human evaluation is un-

avoidably subjective, we erred on the side of caution and only selected terms which could unmistakably be used in a hateful manner. Unfortunately, this terminology extraction strategy has the drawback of not directly taking into account terms that get resemanticized and assume a new, offensive meaning. Further work could be carried out to identify such terms in order to have a more comprehensive understanding of the issue.

With relation to *Il forum dei brutti*, we once again studied terms we deemed to be characteristic of the forum's incel language; however, in this case we focused on 10 terms used to describe other men in negative or positive ways. We chose this approach because the goal of this modern diachronic study is to show that language specific to incels changes over time, regardless of whether it can be considered hateful. Therefore, since in `IFU-22-IT` we could not find as much misogynous or racist jargon as in `IFU-22-EN`, we decided to consider the way men are represented, instead of women.

In order to conduct the study, the contents posted on the *Incels.is* forum from 2017 to 2022 were divided into 22 chronological partitions, one for each 100 pages, each page containing 100 threads. With a similar approach, *Il forum dei brutti* was divided chronologically by grouping posts by year of creation, from 2009 to 2022, for a total of 14 partitions.

The keyness of each selected term was measured for every partition, calculating the slope m of its regression line as:

$$m = \frac{\sum_{i=1}^{n}(t_i - \bar{t})(k_i - \bar{k})}{\sum_{i=1}^{n}(t_i - \bar{t})^2} \tag{4.1}$$

where t_i is the i-th time partition, k_i is the i-th keyness score, n is the number of partitions, and $\bar{t}$ and $\bar{k}$ are the means of the two variables. By calculating the slope of the regression line, we are able to find how the keyness of a term changes over time. A positive slope indicates that the use of a term is becoming more frequent, while a negative slope indicates that a term is becoming less prevalent. For each term, the slope was first calculated across all partitions (22 for *Incels.is* and 14 for *Il forum dei brutti*); then, it was divided by the average keyness of the term over all the partitions, thus obtaining the normalized slope. This was done because certain terms may have very high keyness values, while other terms may not be as prevalent, and we wanted to be able to compare the slope of different terms regardless of the absolute value of their keyness.

For each partition, only the keyness of the 500 terms with the highest

keyness was recorded. Zero values, produced whenever the item's keyness was not high enough to appear among the top 500 terms of the partition, were ignored both for the calculation of the slope and the average keyness. The number of zero values for *Incels.is* was 7.16% of the total, while for *Il forum dei brutti* it was 44.44%.

With relation to *Incels.is*, we selected the 10 terms with the highest and lowest normalized slope, 20 in total, while for *Il forum dei brutti* we only picked the top and bottom 5 terms, 10 in total. The lower number of terms for *Il forum dei brutti* is due to the fact that we could not identify enough relevant terms for the study. For both forums, the mean normalized slope was finally calculated for each group of terms to have a pair of values which could be used to compare the two overall trends.

Since, to the best of our knowledge, this method of analyzing keyness to study the evolution of lexicon in a modern diachronic study is novel, it would be desirable to carry out further experiments to verify the validity of this approach.

4.2 Unsupervised Datasets

Having crawled and scraped the contents of *Incels.is* and *Il forum dei brutti*, we compiled their contents into two dumps, which we use as unsupervised datasets throughout the rest of the study, including Chapter 5. The datasets are organized as lists of posts, each containing the content and metadata listed in Section 4.1.[5]

We refer to the dataset obtained from the dump of the *Incels.is* forum as `IFU-22-EN` (Incel Forum 2022 English Corpus). The posts it contains were collected by crawling the "Inceldom Discussion" section of `https://incels.is/` up to 18 October, 2022. The raw collection contains a total of $4.76M$ posts, divided among $230k$ threads. The dataset extracted from *Il forum dei brutti*, which we refer to as `IFU-22-IT` (Incel Forum 2022 Italian Corpus), was collected by crawling the "Una vita da Brutto" section of `https://ilforumdeibrutti.forumfree.it` up to 4 December, 2022. The content dump is made up of $638k$ posts, organized in $30k$ threads.

The statistics of the two unsupervised datasets are summarized in Table 4.1. As the data shows, the average length of the posts is much longer in

[5]Full datasets in CSV format: `https://zenodo.org/record/7879341`

Table 4.1: Statistics of the two unsupervised datasets, including the mean and standard deviation of the length in tokens of the posts contained in the dataset. The last column shows the median time difference in seconds between the timestamp of the first and second post in each thread.

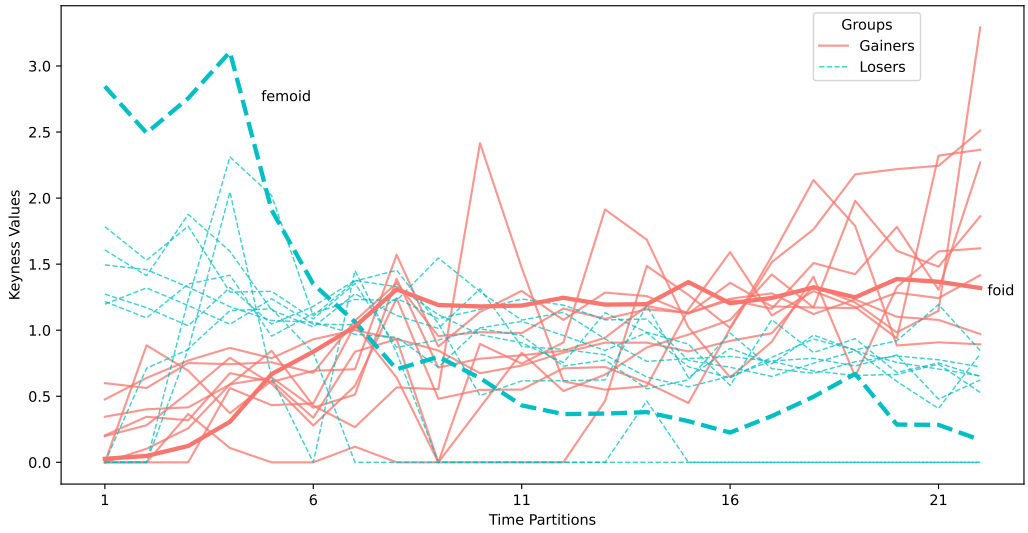
Dataset	**Posts**	**Threads**	**Length**	$\mathbf{t}_{median}$
`IFU-22-EN`	4,756,882	222,965	31.07 ± 70.01	155.00
`IFU-22-IT`	638,143	29,646	52.78 ± 80.77	540.00

`IFU-22-IT`, compared to `IFU-22-EN`. The median posting time difference between an original post and its first response is also much higher in `IFU-22-IT`, with a median of 540 against only 155 seconds. This could hint at the fact that threads in *Il forum dei brutti* are less active as far as the frequency of replies is concerned, but hosting conversations which are more akin to actual discussions, rather than the more chaotic back-and-forths which seem to take place in *Incels.is*.

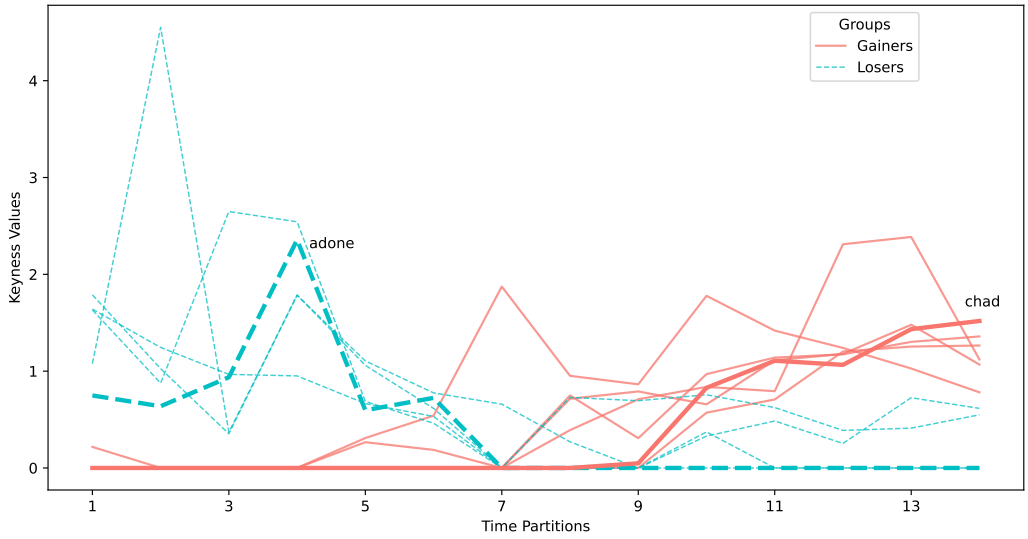
4.3 Diachronic Study Results

Figure 4.2 shows the over-time trend of the keyness of the terms extracted from *Incels.is* and *Il forum dei brutti* over the partitions of the two forums. The curves show clear opposite trends for the two groups, which we refer to as "gainers" and "losers" of keyness, based on whether their mean normalized slope is positive or negative, respectively. The plots help visualize a widening over-time difference in lexicon, which may cause models trained on dated texts to become increasingly worse at evaluating more recent data. The highlighted terms in the figure also show that certain terms seem to substitute each other over time, although not all of them can be paired in this manner. For example, "foid" is a contraction of "femoid" and "adone" is a close synonym of "chad", and for both pairs we can observe opposite trends with a specific point in time in which one overtakes the other.

Table 4.2 reports the normalized slopes of the terms obtained from the two forums. In both cases, the mean normalized slopes of the two data series, compared side by side, quantitatively display a clear trend according to which certain terms gain popularity over time, while others become less popular. With regard to *Incels.is*, the difference between the mean normalized slopes is



(a) *Incels.is*



(b) *Il forum dei brutti*

Figure 4.2: Keyness over time for the characteristic incel terms extracted from the (a) *Incels.is* and (b) *Il forum dei brutti* forums. Red lines represent the terms that gained keyness over time, while blue lines represent the terms that lost keyness over time.

Table 4.2: Keyness normalized slopes for *Incels.is* and *Il forum dei brutti*.

Forum	Gainers		Losers	
	Lexical Term	**Slope**	**Lexical Term**	**Slope**
Incels.is	shitskin	0.093	racepill	-0.019
	deathnic	0.081	stacie	-0.022
	cumskin	0.079	jb	-0.027
	noodlewhore	0.077	chadlite	-0.029
	slav	0.068	whitecels	-0.032
	foid	0.058	cunt	-0.036
	curryland	0.051	slut	-0.046
	aryan	0.048	deathnik	-0.047
	ricecel	0.047	roastie	-0.051
	whore	0.025	femoid	-0.124
	Mean	**0.063**	**Mean**	**-0.043**
Il forum dei brutti	zerbini	0.104	reietto	-0.142
	normie	0.121	strafigo	-0.122
	bv	0.125	figaccione	-0.122
	chad	0.126	attraente	-0.113
	subumano	0.158	adone	-0.103
	Mean	**0.127**	**Mean**	**-0.120**

0.106, while for *Il forum dei brutti* the difference is even larger, 0.247, which points at an even faster lexical evolution. In both cases, the shift in lexicon needs to be taken into account in order to have a clear picture of the language adopted by each speech community. For terms such as "foid", "femoid", and "roastie", the observed trends also confirm the time-series data discussed in Gothard (2020), which show certain terms increasing and decreasing in use over the total messages posted in incel subreddits.

With relation to *Incels.is*, as already anticipated through Figure 4.2, although terms like "foid" and "femoid" have the same meaning (both are used to dehumanize women by associating them to insentient androids[6]), the shorter form has become more popular, while the use of the full form has decreased. This is probably due to the fact that, given the high frequency with which the term is used in the forum, users tend to use the abbreviated version to save time and effort. This might seem like a minor detail, but the

[6] `https://incels.wiki/w/Femoid`

sheer amount of misogyny that is expressed in the forum through this term alone makes it important to point out a shift in its use.

As regards *Il forum dei brutti*, we can observe that the way users refer to men changes in a rather clear way. On one hand, positive words that are commonly used in general language, such as "strafigo" and "figaccione" (both meaning "extremely handsome"), are substituted by specialized terms that are more specific to the forum's speech community, e.g., "chad".[7] On the other hand, we can see the same phenomenon for negative words, where "reietto" ("outcast") loses popularity, leaving space to terms with more specialized uses, such as "bv", meaning "brutto vero" (lit. "truly ugly"), and "subumano", meaning "subhuman". The first is an acronym, which makes its meaning opaque to outsiders, while the second is a term with a much stronger and denigrating connotation.

Based on the conducted qualitative and quantitative analyses, the same conclusions can be drawn for both forums: the presented terms are arguably characteristic of the incel language used within the two platforms and the change in their usage over time is non-negligible. This implies that language models could become progressively worse at predicting over these domains, were their training resources not be periodically updated. Models rely on training material to learn language, and if the material is outdated, their understanding of the discourse currently produced by a specific speech community could become suboptimal. This is especially important considering the fact that, especially in the case of *Incels.is*, the presented racist and misogynous terms are novel and carry most of the discriminatory meaning through neologisms.

Consequently, it seems desirable, if not necessary, to periodically update corpora to have accurate terminological representations. In some cases, it would arguably make sense to even rebuild resources from scratch, were they too outdated. In our case, given the observed changes in keyness, we estimate that the hereby analyzed time frame could be taken as a reference for how long resources can be considered up-to-date. However, with the aim of obtaining an objective figure, further research could be conducted to quantify how often resources should be updated to keep up with the evolution of the language used in the spaces scrutinized through this study.

The necessity to build such material is also supported by the fact that, as discussed in Chapter 3, resources on the topic of incels are rare and limited,

[7] `https://incels.wiki/w/Chad`

and their applicability is often compromised because the linguistic domain of the source data only partially aligns with the one under investigation (Pelzer et al., 2021). An additional cause for such incompatibility of resources can be found in the annotation scheme, which can be inapplicable to the supervised task being approached (Zhou et al., 2022). However, the necessity to build new resources does not mean they will be obsolete soon after being employed, as the time frames we have analyzed in this chapter span various years of forum activity.

4.4 Diachronic Study Contributions

In this chapter we have presented novel sources of genuine data for the study of incel language, in the form of two popular incel online forums, especially in the case of *Il forum dei brutti*, which has not yet been studied in the literature. In addition, we have crawled and scraped the forums, from which we built `IFU-22-EN` and `IFU-22-IT`, two unsupervised datasets containing forum posts from two forums frequented by incels, *Incels.is* and *Il forum dei brutti*. These two datasets, which we release publicly, are organized by post and thread, and contain all the metadata provided by the forum. They can be analyzed manually via corpus linguistics methods, or automatically, via computational linguistics methods, to study the language features of the two incel speech communities. Lastly, we carried out a modern diachronic study of the keyness of the terms characteristic of incel language, showing the desirability of building updated resources for its analysis.

Chapter 5

Experiments

This chapter lays out the NLP-oriented experiments conducted in the study. Section 5.1 describes the annotation process of the datasets used in the experiments. Section 5.2 lays out the experimental settings approached in the study. Section 5.3 introduces the hate speech detection experiments and its results. Section 5.4 presents the approach used for the multi-label setting, which approaches the task of racism and misogyny detection, and reports the results obtained. Section 5.5 illustrates the forecasting setting of the study and reports its results. Section 5.6 summarizes the contributions of the chapter.

5.1 Dataset Annotation

Supervised datasets in English and Italian were obtained from `IFU-22-EN` and `IFU-22-IT` by annotating their posts. They were built for the purpose of training models for the identification of 1) hate speech and 2) misogyny and/or racism. To this end, user posts were annotated with two independent binary labels, one for misogyny and one for racism.

We refer to the dataset obtained from `IFU-22-EN` as `IFS-EN` (Incel Forum Supervised, English). The English dataset was initially sampled with two constraints: 50% of the posts had to include at least one characteristic term from the incel jargon shown in Table 4.2, while the other 50% was sampled so that it contained no such terms. In addition, instances had to be longer than five words. The former constraint sought to balance the occurrence of instances with and without incel jargon to prevent models from overly

Please identify whether each post is categorized as misogynous, racist, or falls into another category:
A post is deemed **misogynous** if it exhibits any of the following traits:

- Objectifies or stereotypes women;
- Claims that men are superior to women;
- Derails the conversation to defend the abuse of women, deny male responsibility, or redirect the conversation in favor of men;
- Contains sexual advances, solicits sexual favors, sexually harasses the recipient, or threatens women with physical violence to assert power;
- Uses slurs against women without any legitimate purpose.

A post is considered **racist** if it demonstrates any of the following traits:

- Uses a racial slur;
- Stereotypes, attacks, or seeks to silence a minority without a valid argument;
- Promotes violent crime against minorities;
- Misrepresents the truth or distorts views on a minority with baseless claims;
- Shows support for problematic ideologies, such as xenophobia, homophobia, or sexism.

Figure 5.1: Guidelines for the corpus annotation, derived from Fersini et al. (2018) for misogyny and Waseem and Hovy (2016) for racism.

relying on it, while the latter aimed at excluding instances which would not be useful during training.

A pilot annotation was first carried out by three annotators on a subset of 50 instances, obtaining an inter-annotator agreement (IAA) (Bobicev and Sokolova, 2017) of 0.77, using Cohen's Kappa. This is considered a "substantial" agreement, 0.81 being the threshold for "almost perfect". Due to the good IAA obtained during the pilot annotation, the rest of the instances were annotated by a single annotator. The annotators are all experts in the relevant subject matter, possessing a strong foundation in linguistics and gender studies, as well as extensive knowledge of NLP and data annotation for developing supervised models. All three annotators have a C2 CEFR level of English. The annotation was carried out following the guidelines shown in Figure 5.1, producing labels whose statistics are reported in Table 5.1. The split between the train, development and test partitions of the dataset is 70/15/15.[1]

[1]Obtained using the *train_test_split* function contained in the SciKit-Learn library and setting the seed to 42.

Table 5.1: Statistics of the IFS-EN and IFS-IT datasets, reporting the shares of instances labeled as hate speech (HS), misogynous (M), racist (R), both or neither.

Dataset	Binary		Multi-Label			
	HS	**Non-HS**	**M**	**R**	**Both**	**Neither**
IFS-EN$_{\texttt{tr}}$	1,482	2,160	806	630	46	2,160
IFS-EN$_{\texttt{de}}$	316	464	173	130	13	464
IFS-EN$_{\texttt{te}}$	292	489	160	125	7	489
IFS-IT	200	300	187	8	5	300

As regards the dataset obtained from IFU-22-IT, we refer to it as IFS-IT (Incel Forum Supervised, Italian). We built it by annotating 500 random instances from IFU-22-IT, excluding messages that were empty, excessively short or lacking in meaning. This time, we did not sample instances with any particular terminology constraints, since our preliminary analysis of the corpus (see Section 4.1) showed that the amount of misogynous and racist jargon was not as prevalent as in IFU-22-EN.

In this case we relied on two annotators which obtained an IAA of 0.69 during the pilot annotation of 50 instances. Once again, we used Cohen's Kappa to calculate the IAA. Both annotators are native speakers of Italian and, just like in the case of IFS-EN, both are experts on the topic at hand. As the IAA was deemed acceptable, the 450 other instances were all annotated by a single annotator. Posts sampled from IFU-22-IT were annotated until the resulting IFS-IT had enough hateful instances to match the ratio of hateful instances contained in IFS-EN, i.e., 40%. Once again, the annotation was carried out by following the guidelines shown in Figure 5.1, producing labels whose statistics are reported in Table 5.1.

5.2 Experimental Settings

The annotation strategy laid out in Section 5.1 allows us to study two different experimental settings directly and a third one derivatively.

Setting 1: The hate speech detection setting consists in a binary classification task, carried out in mono- and cross-lingual scenarios. Here, we

use labels produced indirectly from the "racist" and "misogynous" labels of `IFU-22-EN` and `IFU-22-IT`, respectively. In particular, we consider a post to be hateful if it is labeled as either misogynous or racist, and non-hateful otherwise. In the monolingual scenario we employ monolingual (English) Transformer models, while in the cross-lingual one we only use multilingual Transformer models. The cross-lingual scenario is zero-shot, meaning that models are tested directly on `IFS-IT` without ever fine-tuning the models on instances from the same language and domain. That is, we are never using forum posts extracted from `IFS-IT` for training. Two strategies are used to improve the performance of models in this setting: dataset merging and MLM pre-training.

Setting 2: The misogyny and racism detection setting consists in a multi-label classification task, also carried out in monolingual and cross-lingual scenarios. A separate binary classification task is approached for each of the two labels originally assigned by the annotators. In the monolingual scenario we approach the task by using monolingual models, while in the cross-lingual zero-shot scenario we only use multilingual models.

Setting 3: The hate speech forecasting setting consists in a regression task, carried out in monolingual and cross-lingual scenarios. The datasets used in this task are obtained by automatically labeling `IFU-22-EN` and `IFU-22-IT` for hate speech by using the best-performing monolingual and multilingual models developed in the hate speech detection task. The datasets are organized in pairs of original posts ("OPs") and hate speech scores, the latter being a value between 0 and 100. Models are trained to predict a scalar value for each OP, which represents the predicted percentage of hateful replies the OP will produce.

5.3 Hate Speech Detection

This section discusses hate speech detection, which represents the binary text classification setting of the study. We approach this task in monolingual and cross-lingual scenarios.

As mentioned in Section 5.1, in `IFS-EN` a post is considered as expressing hate speech if it is labeled as being either racist or misogynous. In this

experimental setting, the objective is to develop a model capable of effectively discerning when a post is hateful. For that, we evaluate the models on IFS-EN's testing partition in the monolingual scenario. In the cross-lingual scenario, we evaluate on IFS-IT, approaching the task in a zero-shot setup. For this binary classification setting, models are evaluated by using precision, recall, and F_1-measure, as introduced in Section 2.5.1.

5.3.1 Dataset Augmentation

In both the mono- and cross-lingual scenarios, we first train a series of models solely using the IFS-EN supervised dataset and evaluate them on $\texttt{IFS-EN}_{\texttt{te}}$ and IFS-IT, respectively. Taking inspiration from Pelicon et al. (2021), we then attempt improving the best-performing model for each scenario by training it on various combinations of supervised datasets annotated for hate speech.

This approach is not always successful, as explained in Pelzer et al. (2021) and Pelicon et al. (2021), since different supervised datasets are annotated following different schema, although they are meant to train a model for the same task. However, as the latter show, if dataset content and annotation scheme match sufficiently well, this method can lead to an improvement in model performance.

We use datasets annotated for hate speech in three languages: English, Italian, and Spanish. The choice of datasets had two constraints: 1) the language had to be similar to either English or Italian, and 2) their annotation schema had to pertain to hate speech, misogyny or racism. By making sure the annotation schema are similar, we ensure the datasets can be merged and used in unison for training. As such, only the datasets listed in Table 5.2 were ultimately considered.[2]

For English, we use a greater number of datasets, compared to Italian and Spanish, since using this approach in the monolingual scenario requires various English-language datasets:

- Davidson et al. (2017): a Twitter dataset with multi-class labels in which posts are annotated as "hate", "offensive" or "neither". In our study, we only use the first "hate" binary annotation.

[2]The published datasets can all be found at: https://github.com/leondz/hatespeechdata. We could not get access to the HASOC German dataset, which we could have also used in our experiments, since German as a language is close to English.

Table 5.2: List of supervised datasets used to train models in the mono- and cross-lingual scenarios of the hate speech detection setting.

Dataset		Source	Language
Davidson	Davidson et al. (2017)	Hatebase.org	en
HateXplain	Mathew et al. (2021)	Twitter+Gab	en
Stormfront	Mathew et al. (2019)	Stormfront.org	en
$\text{HatEval}_{\text{en}}$	Basile et al. (2019)	Twitter	en
HSD_{fb}	Bosco et al. (2018)	Facebook	it
HSD_{tw}	Bosco et al. (2018)	Twitter	it
AMI_{20}	Fersini et al. (2020)	Twitter	it
$\text{HatEval}_{\text{es}}$	Basile et al. (2019)	Twitter	es

- Stormfront(Mathew et al., 2019): a dataset annotated for hate speech, gathered from the Stormfront forum, a white nationalist community often characterized by racist discussions.
- $\text{HatEval}_{\text{en}}$ (Basile et al., 2019): the English portion of the HatEval 2019 Twitter dataset for the detection of hate speech against migrants and women. Since the datasets are neatly split 50/50 for the two categories, we only use the half of the dataset pertaining to misogyny detection, for a total of 6,500 instances. The rationale behind this choice was that we considered the instances annotated for hate speech against migrants not to be relevant with regard to incel speech.
- HateXplain (Mathew et al., 2021): a dataset gathered from Gab and Twitter, containing multi-class labels obtained through majority vote for three classes: "normal", "offensive", or "hate". It also contains attention masks for explainability, but we only consider the hate speech annotation and use it as a binary label, taking into account the majority vote among the three annotators.
- IFS-EN: see Section 5.1

For Italian, we use the 2018 Hate Speech Detection (HaSpeeDe) task Facebook and Twitter datasets (which we refer to as "HSD_{fb}" and "HSD_{tw}", Bosco et al. (2018)), and the 2020 Automatic Misogyny Identification dataset ("AMI_{20}", Fersini et al. (2020)). The first are binary-annotated for hate speech, while the second dataset is binary-annotated for misogyny. As previously stated, we also use IFS-IT for evaluation in the cross-lingual scenario.

For Spanish, we use the $6.6k$ annotated tweets from the Spanish-language

portion of HatEval 2019 ("HatEval$_{\texttt{es}}$", Basile et al. (2019)), which uses the same annotation scheme as its English counterpart.

The training, development, and testing sets for the Davidson, HateXplain and Stormfront datasets were obtained by partitioning the original datasets with a 70/15/15 split. For the rest of the datasets, we used the original partitions provided by the authors.

5.3.2 Monolingual Binary Setting

In the monolingual scenario, we first attempt improving the performance of the models via monolingual MLM pre-training, thanks to which we develop new models. Subsequently, we test the newly-obtained models on IFS-EN and pick the best-performing one, attempting to further improve its performance by using various combinations of English datasets for fine-tuning.

We use BERT$_{base}$ (Devlin et al., 2019) and RoBERTa$_{base}$ (Liu et al., 2019) as our baselines. We also use HateXplain (Mathew et al., 2021), a model already fine-tuned for hate speech detection, and HateBERT (Caselli et al., 2021), a BERT$_{base}$ model with MLM pre-training done on RAL-E, a large-scale dataset of Reddit comments in English from communities banned for being offensive, abusive, or hateful.

5.3.2.1 Monolingual MLM Training

Based on the approach used by Caselli et al. (2021), we attempt improving BERT$_{base}$ and RoBERTa$_{base}$'s understanding of the incel language by training them on the MLM task, as described in Section 2.4.2.1. Three unsupervised datasets in total are prepared for this step, built by subsampling random instances from the IFU-22-EN corpus ($10k$, $100k$, and $1M$ posts). None of the instances used for MLM pre-training include data from IFS-EN and IFS-IT. For each model, the sentences are first tokenized using HuggingFace's AutoTokenizer,[3] which automatically selects the appropriate tokenizer for the model we wish to train. The sentences are then fed into the model using HuggingFace's data collator for language modeling,[4] which automatically masks tokens with a 15% chance for the MLM task. Finally, the models are

[3] https://huggingface.co/docs/transformers/model_doc/auto#transformers.AutoTokenizer

[4] https://huggingface.co/docs/transformers/main_classes/data_collator

Table 5.3: Results for the monolingual hate speech text classification setting, only using IFS-EN's partitions for training, development and testing.

Model	(e)	Validation			Test		
		F1	Rec	Prec	F1	Rec	Prec
BERT_{base}	(3)	0.846 ± 0.010	0.851	0.845	0.845 ± 0.008	0.843	0.849
I-BERT_{10k}	(4)	0.867 ± 0.005	0.870	**0.865**	0.865 ± 0.008	0.855	**0.876**
I-BERT_{100k}	(3)	0.865 ± 0.006	0.887	0.846	0.868 ± 0.006	0.882	0.855
I-BERT_{1M}	(4)	**0.875 ± 0.005**	**0.894**	0.856	**0.872 ± 0.006**	**0.883**	0.861
RoBERTa_{base}	(4)	0.851 ± 0.007	0.857	0.845	0.841 ± 0.005	0.851	0.831
I-RoBERTa_{10k}	(4)	0.856 ± 0.005	0.870	0.842	0.843 ± 0.005	0.863	0.824
I-RoBERTa_{100k}	(4)	0.864 ± 0.008	0.870	0.858	0.844 ± 0.005	0.853	0.836
I-RoBERTa_{1M}	(4)	0.860 ± 0.005	0.864	0.857	0.857 ± 0.005	0.878	0.837
HateBERT	(4)	0.853 ± 0.007	0.845	0.861	0.853 ± 0.008	0.849	0.857
HateXplain	(3)	0.856 ± 0.005	0.854	0.859	0.847 ± 0.005	0.836	0.859

trained for one epoch with a batch size of 32 samples on a single Tesla P100 GPU with 16 GB of VRAM. We refer to the two model types obtained from this process as "I-BERT" and "I-RoBERTa" (short for "Incel BERT" and "Incel RoBERTa").

5.3.2.2 Monolingual MLM Training Results

We train each model five times using $\texttt{IFS-EN}_{\texttt{tr}}$ and select the number of epochs based on the performance achieved on $\texttt{IFS-EN}_{\texttt{de}}$. We do this in order to make our results more reliable and diminishing the effect of the random initialization of the models. The resulting models are then evaluated on $\texttt{IFS-EN}_{\texttt{te}}$. Table 5.3 reports the results for the monolingual MLM training experiment.

As far as the BERT_{base} and RoBERTa_{base} models are concerned, we observe that the MLM pre-training strategy improves the performance of both models on $\texttt{IFS-EN}_{\texttt{te}}$, after fine-tuning is carried out. With relation to BERT, training on $1M$ sentences leads to an average improvement of 2.7 F_1 points (3.20% increase) for I-BERT_{1M}, making it the best model overall in the monolingual scenario. As far as RoBERTa is concerned, training it on $1M$ sentences leads to an average improvement of 1.6 F_1 points (1.06% increase) for I-RoBERTa_{1M}.

Additionally, we can see that an increasing amount of MLM training leads to better performance for both BERT_{base} and RoBERTa_{base}. In the case of BERT_{base}, it is also interesting to notice that even a very small amount of

Table 5.4: Results for the binary monolingual text classification setting for the random initializations of BERT_{base} and I-BERT_{1M}.

Model	Validation			Test		
	F1	**Rec**	**Prec**	**F1**	**Rec**	**Prec**
BERT_{base}	0.419 ± 0.222	0.634	0.389	0.397 ± 0.206	0.634	0.368
I-BERT_{1M}	0.382 ± 0.194	0.471	0.479	0.376 ± 0.181	0.488	0.364

training leads to a noticeable improvement. By comparison, with a training effort of two fewer magnitudes, compared to training on a million instances for I-BERT_{1M}, I-BERT_{10k} already obtains a mean F_1 score which is sizeably better than BERT_{base}'s (+2 points), reaching performances comparable to those of I-BERT_{100k} (+2.3) and I-BERT_{1M} (+2.7). As regards RoBERTa_{base}, the improvement obtained through this strategy is not as substantial, with an increase of 0.2 points for I-RoBERTa_{10k}, 0.3 points for I-RoBERTa_{100k} and 1.6 points for I-RoBERTa_{1M} over the baseline. Since RoBERTa_{base} was trained on a larger amount of data and for a longer time compared to BERT_{base}, the smaller performance boost might be due to the fact that biasing the RoBERTa model towards a new domain might require more data and training, compared to BERT.

Table 5.4 reports the results for the random initializations of the BERT models, which show that any improvement in performance is always obtained only after fine-tuning. The mean F_1 scores in this case are substantially lower than when fine-tuning the models on `IFS-EN` and the standard deviation is also very high, indicating not just poor performance, but also high model instability.

The boost in performance obtained through this strategy can be taken as evidence that teaching models about the characteristic features of the language used in *Incels.is* is ultimately important. This fact further supports the study conducted in Chapter 4, as it shows that the language used in *Incels.is* is indeed different from the language used in general English, when it comes to expressing misogynous and racist sentiments. The results obtained even with a moderate amount of training data, as few as 10,000 sentences, also allow us to conclude that the difference in language may be very easily learned by models. This could be due to the fact that, while the language used in *Incels.is* is indeed different from the language used in the general population, the differences in the expression of hate speech are due to the presence of a small number of novel words and words which are used with

novel meanings.

However, as already mentioned at the end of Section 4.1, not all words used with novel meanings can be spotted through the keyword method adopted in Chapter 4. This means the study is limited in this regard, opening up the possibility for further research, which would have the goal to find a way to spot terms whose meaning deviates from the one usually attributed to them. This could help further verify whether the language used in *Incels.is* is indeed different from the language used by the general population, both through corpus linguistic methods and the use of NLP models.

5.3.2.3 Monolingual Baseline Dataset Fine-Tuning

After improving the performance of the baseline Transformers via MLM pre-training, we proceed with fine-tuning and evaluating the models solely on `IFS-EN`'s partitions. The results obtained in this setting represent the baseline against which the dataset merging experiment, laid out in the next subsection, is compared.

For all Transformer models, prior to training, the sentences contained in `IFS-EN` are tokenized with a maximum length of 256 tokens, padding them to max length, including [CLS] tokens and returning attention masks. All Transformer models are trained for four epochs with a batch size of 16, using the AdamW optimizer with a learning rate of 10^{-5} and an epsilon of 10^{-8}. We log metrics for all epochs over five training runs, in order to find out at what epoch the models are on average getting the best results. Each run, the models are initialized with a different seed.

5.3.2.4 Monolingual Dataset Merging

After evaluating all the aforementioned models on $\texttt{IFS-EN}_{\texttt{te}}$, we rank them based on the obtained F_1 scores. We then pick the top performer (i.e., I-BERT$_{1M}$, henceforth referred to as simply "I-BERT") and attempt further improving it by training it on various combinations of the datasets described in Section 5.3.1. For some dataset combinations, we subsample the training partitions so that all the training sets being merged have the same number of instances. For this part of the experiment, we use BERT$_{base}$ as our baseline, to observe the effect of this strategy without MLM pre-training. While training on these dataset combinations, we use the same pre-processing and training parameters as the ones used when training solely on $\texttt{IFS-EN}_{\texttt{tr}}$.

Table 5.5: Results for the monolingual hate speech detection task, after fine-tuning on different combinations of datasets (■ = full, □ = sub-sampled).

	Datasets				e	Validation (IFS-EN$_{de}$)			Test (IFS-EN$_{te}$)		
	Davidson	HateXplain	Stormfront	HatEvalen		F_1	Rec	Prec	F_1	Rec	Prec
BERT$_{base}$					3	0.846±0.010	0.851	0.845	0.845±0.008	0.843	0.849
	■				4	0.838±0.010	0.834	0.843	0.851±0.006	0.852	0.849
				□	3	0.858±0.007	0.858	0.859	0.844±0.008	0.835	0.853
	□			□	4	0.848±0.007	0.846	0.851	0.840±0.011	0.840	0.840
				■	2	0.857±0.007	0.874	0.840	0.837±0.007	0.854	0.820
		■			4	0.854±0.007	0.857	0.851	0.842±0.007	0.842	0.842
		■		■	4	0.858±0.005	0.858	0.857	0.845±0.007	0.841	0.848
			■		4	0.853±0.008	0.854	0.852	0.855±0.005	0.863	0.848
			■	■	3	0.847±0.002	0.853	0.843	0.849±0.009	0.862	0.837
I-BERT					4	**0.875±0.005**	**0.894**	0.856	**0.872±0.006**	0.883	0.861
	■				2	0.864±0.005	0.877	0.850	0.857±0.005	0.879	0.836
				□	4	0.886±0.004	0.908	0.866	0.852±0.003	0.860	0.844
	□			□	2	0.869±0.007	0.887	0.851	0.845±0.005	0.874	0.818
				■	1	0.866±0.003	0.878	0.854	0.855±0.003	0.877	0.833
		■			1	0.858±0.003	0.789	**0.940**	0.857±0.008	0.804	**0.918**
		■		■	3	0.875±0.006	0.891	0.860	0.856±0.008	0.875	0.838
			■		4	0.859±0.004	0.861	0.858	0.865±0.004	**0.884**	0.848
			■	■	3	0.859±0.002	0.882	0.838	0.859±0.002	0.882	0.838

5.3.2.5 Monolingual Dataset Merging Results

Table 5.5 reports the results for the monolingual dataset merging experiment. The data includes validation and test mean F_1-measure, recall and precision for the epoch (e) at which the mean validation F_1-measure is highest. We log metrics over five training runs for each model and dataset combination, along with the standard deviation of the F_1 scores. All training combinations contain IFS-EN$_{tr}$ and are evaluated on IFS-EN$_{de}$ and IFS-EN$_{te}$. The models are initialized with a different seed each run.

Combining IFS-EN$_{tr}$ with the Stormfront, Davidson, and [Stormfront, HatEval$_{en}$][5] datasets slightly improves BERT's performance, yielding an improvement of 1, 0.6 and 0.4 points on IFS-EN$_{te}$, respectively. Neither HatEval$_{en}$ nor HateXplain contribute positively, when used by themselves

[5]This use of the square brackets represents the union of multiple datasets.

or together, as [$\texttt{HatEval}_{\texttt{en}}$, HateXplain].

In the case of $\texttt{HatEval}_{\texttt{en}}$, this is probably due to the fact that the dataset is only focused on misogynous hate speech, which is not entirely representative of the problem at hand. In addition, the language used in IFS-EN and $\texttt{HatEval}_{\texttt{en}}$ is rather different, since the English 2019 HatEval dataset is built on Twitter, which mostly comprises only general-language misogyny, while IFS-EN contains many language features specific to the incel community. Still, this result is interesting, as the majority of hate speech in IFS-EN is conveyed through misogyny, meaning that a dataset built to train a model to detect misogyny should in theory help in detecting misogynous hate speech also in IFS-EN.

As far as HateXplainis concerned, the dataset most likely failed to improve the performance of the model because it was built to be used jointly with the attention arrays it contains and because its sentences are already tokenized and stripped of punctuation, which means the model has less syntactical information to work with.

As for I-BERT, all combinations yielded worse results than the baseline. This could be due to the fact that the model became too biased toward $\texttt{IFS-EN}_{\texttt{tr}}$ during MLM pre-training, making it unable to learn effectively from other datasets. That said, its performance on $\texttt{IFS-EN}_{\texttt{te}}$ is still better than the performance BERT achieves when merging $\texttt{IFS-EN}_{\texttt{tr}}$ with the Stormfront, Davidson, or [Stormfront, $\texttt{HatEval}_{\texttt{en}}$] datasets.

Consequently, in this setting of the study, MLM pre-training as a model-improvement strategy outperforms fine-tuning models on combinations of different datasets. However, despite the boost in performance offered by this strategy being lower compared to the MLM results reported in Section 5.3.2.2, the result is still significant, as it hints at the fact that the provided resource also has the potential to be employed jointly with other previously released datasets. This means it could become a valuable tool for future research not only as far as incel hate speech is concerned, but also for the broader field of hate speech detection in general.

5.3.3 Multilingual Binary Setting

In the cross-lingual scenario, we use mBERT (Devlin et al., 2019) as our baseline model, whose performance we first attempt improving via multilingual MLM training, developing new pre-trained models. Then, we test the

newly-obtained models on IFS-IT and pick the best performer, attempting to further improve its performance by using various multilingual combinations of datasets for training.

5.3.3.1 Multilingual MLM Training

We obtain additional models by adopting the same MLM pre-training process we used for BERT$_{base}$ and RoBERTa$_{base}$, in accordance with Caselli et al. (2021). In this case, however, we use both IFU-22-EN and IFU-22-IT. In a similar manner to the monolingual scenario, we use three subsets of sizes $10k$, $100k$, and $1M$ instances. We also use three language combinations: English, Italian, and English + Italian (EN-IT). When using monolingual English data we obtain the three subsets of the two forums by sampling, respectively, $10k$, $100k$ and $1M$ instances from IFU-22-EN. For Italian data, we sample $10k$, $100k$ instances from IFU-22-IT and its full length of $627k$ instances. When using EN-IT bilingual data, we obtain the three subsets of the two forums by sampling, respectively, $5k$, $50k$ and $500k$ instances from IFU-22-EN and IFU-22-IT, with a 50/50 split between English and Italian instances. Just like in the monolingual scenario, none of the instances present in IFS-EN and IFS-IT are used for the MLM pre-training task.

Prior to MLM training, we tokenize sentences using BERT's own tokenizer, BertTokenizer.[6] Then, we feed the data into the model using HuggingFace's data collator for language modeling, which automatically produces token maskings with a 15% probability. Finally, each model is trained for one epoch with a batch size of 32. This way, we obtain three new versions of the base mBERT model, which we refer to as "I-mBERT" (short for "Incel mBERT"), followed by the number of instances they were trained on, for a total of nine new models.

5.3.3.2 Multilingual MLM Training Results

We train each model 10 times using IFS-EN$_{\texttt{tr}}$ and select the number of epochs based on the performance on the IFS-EN$_{\texttt{de}}$ development set. We do this in order to make our results more reliable and diminish the effect of the random initialization of the models. The resulting models are then evaluated

[6] https://huggingface.co/docs/transformers/model_doc/bert#transformers.BertTokenizer

Table 5.6: Validation results for the baseline mBERT model and the "I-mBERT" model variations in the binary cross-lingual text classification setting. Column (e) refers to the number of training epochs, based on validation performance. The best scores are highlighted in bold.

		Validation (IFS-EN$_{de}$)		
Model	**(e)**	**F_1**	**Recall**	**Precision**
baseline	(2)	0.843 ±0.005	0.862	0.826
IT-10k	(4)	0.842 ±0.005	0.868	0.818
IT-100k	(4)	0.847 ±0.005	0.862	0.834
IT-627k	(4)	0.844 ±0.006	0.855	0.834
EN-10k	(4)	0.854 ±0.006	0.882	0.827
EN-100k	(2)	0.852 ±0.003	0.876	0.830
EN-1M	(4)	0.859 ±0.006	0.882	0.837
EN-IT-10k	(2)	0.847 ±0.009	0.863	0.833
EN-IT-100k	(4)	0.852 ±0.007	0.882	0.825
EN-IT-1M	(4)	**0.863 ±0.004**	**0.887**	**0.841**

on IFS-IT, and on IFS-EN$_{te}$ for reference. Table 5.6 reports the validation results for the multilingual MLM pre-training experiments, while Table 5.7 reports the test results.

As regards monolingual MLM training, the validation performance fluctuates both in the English validation and test sets, still ending up over the baseline when using the full amount of training data. On the Italian test set, using a small amount of training data initially produces a performance boost for the $10k$ versions, but adding more data leads to a performance drop. When using English data, this could be due to the fact that the model becomes better at learning from the English training data, but grows farther from the Italian test set. When using Italian data, the reason could be the opposite, i.e., the model becomes slightly better at predicting, but worse at learning from the English training data to the point where the overall performance drops.

Conversely, carrying out MLM on mBERT with a small amount of bilingual data ($10k$ instances) initially hinders the performance of the model. However, the performance eventually improves over the baseline given enough bilingual data ($100k$ and $1M$ instances). This could be due to the fact that the capability of the mBERT model to understand syntactic and seman-

Table 5.7: Test results for the baseline mBERT model and the "I-mBERT" model variations in the binary cross-lingual text classification setting. Column (e) refers to the number of training epochs, based on the performance on validation. The best scores are highlighted in bold. We only show Std Dev for the F_1-measure for reasons of space.

		Test (IFS-EN$_{te}$)			**Test** (IFS-IT)		
Model	**(e)**	**F_1**	**Rec**	**Prec**	**F_1**	**Rec**	**Prec**
baseline	(2)	0.826±0.007	0.803	0.851	0.333 ± 0.114	0.224	0.742
IT-10k	(4)	0.840±0.009	0.807	0.876	0.410 ± 0.099	0.290	0.746
IT-100k	(4)	0.836±0.007	0.809	0.865	0.249 ± 0.089	0.150	0.804
IT-627k	(4)	0.836±0.008	**0.819**	0.855	0.111 ± 0.060	0.060	0.861
EN-10k	(4)	0.837±0.005	0.797	0.881	0.501 ± 0.050	**0.378**	0.762
EN-100k	(2)	0.835±0.009	0.797	0.878	0.371 ± 0.106	0.246	0.843
EN-1M	(4)	0.835±0.005	0.789	0.888	0.112 ± 0.034	0.060	0.857
EN-IT-10k	(2)	0.831±0.004	0.806	0.858	0.179 ± 0.060	0.102	0.831
EN-IT-100k	(4)	0.824±0.007	0.783	0.871	0.341 ± 0.079	0.221	0.793
EN-IT-1M	(4)	**0.845±0.006**	0.801	**0.894**	**0.503 ± 0.042**	0.356	**0.864**

tic language relations is initially thrown off with respect to the initial pre-training. The best overall performance is achieved by training mBERT on $1M$ bilingual instances on the fourth epoch of training, with a mean F_1 score of 0.503. Compared to the baseline mBERT model, which achieves a mean F_1 score of 0.333, this represents a significant performance boost of 17 points, showing the effectiveness of this approach.

While the amount of monolingual data used to conduct MLM training is inversely proportional to the performance of the model on the Italian test set, multilingual data exhibits a direct proportionality. On one hand, using English data seems to excessively bias the model towards the training data, with a consequent drop in performance on the Italian test data. On the other, using Italian data overly biases it towards the test set language, hindering training capabilities in English. This is not the case when using bilingual data, which could be due to the fact that exposing the model to both language domains strikes a balance between being able to learn from the training data and generalizing what it has learned to the test data.

5.3.3.3 Multilingual Dataset Merging

Subsequently, we pick the model which performed best on IFS-IT (i.e., I-mBERT$_{EN\text{-}IT\text{-}1M}$, henceforth referred to as simply "I-mBERT") and attempt further improving its performance by training it on combinations of the datasets listed in Table 5.2. The number of epochs is selected based on the performance on IFS-EN$_{\texttt{de}}$. We also run this experiment using mBERT$_{base}$ to obtain baseline metrics and examine the impact of this strategy without the involvement of MLM pre-training. Training is done using the same text preprocessing and training parameters described in the previous paragraphs.

5.3.3.4 Multilingual Dataset Merging Results

Table 5.8 displays the results for the multilingual dataset merging experiments. All training combinations contain IFS-EN$_{\texttt{tr}}$ and are evaluated on IFS-EN$_{\texttt{de}}$ and IFS-IT. For each combination, the highest mean F_1 score over five runs is reported, along with its standard deviation and the epoch (e) at which the results were obtained. The models are initialized with different seeds for each run. As the data shows, using multilingual combinations of different datasets yielded results far above the baseline in almost all cases when evaluating on IFS-IT, both for mBERT and I-mBERT.

As regards mBERT, used as the baseline for this setting, all multilingual dataset combinations improved the model, compared to training solely on IFS-EN$_{\texttt{tr}}$. The best results were obtained using HSD$_{\texttt{fb}}$, [HSD$_{\texttt{fb}}$, HSD$_{\texttt{tw}}$] and HSD$_{\texttt{tw}}$(i.e., the EVALITA 2018 HaSpeeDe datasets). The fact that the best performance is achieved using HSD$_{\texttt{fb}}$ suggests great affinity between its annotation scheme and the one used for IFS-IT. We can see that the model trained using HSD$_{\texttt{fb}}$ obtains a mean F_1 score of 0.694, while for the baseline the achieved mean F_1 score is 0.333. This is a substantial increase in performance of 36.1 points. It is also interesting to notice that the standard deviation of the model is more than nine times lower compared to the baseline when adding HSD$_{\texttt{fb}}$, suggesting that the model, along with becoming better at predicting instances, also becomes substantially more stable.

In analyzing this result, it is important to remember that the EVALITA datasets are built specifically for the detection of hate speech against women, i.e., misogyny identification. Additionally, most of the hate speech contained in IFS-IT is expressed in terms of misogyny, rather than having a more even split between misogyny and racism, like IFS-EN. This is easily verifiable

Table 5.8: Results for the performance of mBERT and I-mBERTon $\texttt{IFS-EN}_{\texttt{de}}$ and IFS-IT after fine-tuning, for different multilingual combinations of datasets. We only show Std Dev for the F_1-measure for reasons of space. The best scores are highlighted in bold.

	EN	IT			ES	e	Validation ($\texttt{IFS-EN}_{\texttt{de}}$)			Test (IFS-IT)		
	HatEval$_{\texttt{en}}$	HSD$_{\texttt{fb}}$	HSD$_{\texttt{tw}}$	AMI$_{20}$	HatEval$_{\texttt{es}}$		F_1	Rec	Prec	F_1	Rec	Prec
mBERT						2	0.843±0.005	0.862	0.826	0.333±0.114	0.224	**0.742**
	■					3	0.847±0.010	0.873	0.823	0.538±0.090	0.435	0.737
					■	4	0.840±0.004	0.866	0.815	0.592±0.026	0.507	0.714
	■				■	2	0.838±0.014	**0.902**	0.784	0.612±0.038	0.545	0.703
		■				2	0.835±0.010	0.837	0.835	**0.694±0.011**	**0.859**	0.583
			■			3	0.854±0.011	0.875	0.835	0.657±0.035	0.721	0.612
		■	■			1	0.825±0.005	0.780	**0.876**	0.690±0.012	0.807	0.605
				■		1	0.825±0.017	0.847	0.804	0.647±0.036	0.687	0.619
		■	■	■		3	**0.850±0.005**	0.839	0.862	0.650±0.015	0.733	0.585
I-mBERT						4	**0.863±0.004**	**0.887**	0.841	0.503±0.042	0.356	**0.864**
	■					4	0.860±0.005	0.868	0.852	0.459±0.047	0.322	0.807
					■	4	0.862±0.007	0.865	0.860	0.669±0.021	0.596	0.765
	■				■	1	0.849±0.002	0.877	0.823	0.689±0.016	0.622	0.776
		■				4	0.862±0.002	0.856	0.867	0.704±0.003	**0.893**	0.582
			■			1	0.859±0.007	0.886	0.834	0.695±0.023	0.641	0.764
		■	■			1	0.855±0.008	0.834	**0.877**	**0.721±0.010**	0.842	0.630
				■		1	0.857±0.008	0.843	0.871	0.623±0.038	0.560	0.708
		■	■	■		1	0.851±0.006	0.877	0.827	0.678±0.014	0.712	0.649

by looking at the proportions of misogyny and racism instances contained in IFS-IT, in which only 1.60% of the instances are annotated as being exclusively racist, as opposed to the misogynous ones representing 37.40% of the total (see Table 5.1). Consequently, using $\texttt{HSD}_{\texttt{fb}}$ and $\texttt{HSD}_{\texttt{tw}}$ could be improving the performance of the model because 1) they are in Italian and 2) their annotation scheme and the hate speech expressed in them is more similar to IFS-IT than IFS-EN. As such, the EVALITA 2018 datasets could work well for training even by themselves, which is why in future work it would be interesting to verify whether training only on combinations of them would yield better results than also adding $\texttt{IFS-EN}_{\texttt{tr}}$, when testing on IFS-IT.

While the performance boost obtained by merging IFS-EN with Italian-

language datasets could be seen as an expected result, the strategy worked also for English and Spanish datasets. Even more interesting is the fact that the performance obtained using [$\texttt{HatEval}_{\texttt{en}}$, $\texttt{HatEval}_{\texttt{es}}$] (both English and Spanish HatEval datasets) is better than using the individual datasets. The performance boost is even more surprising when we consider the fact that the Spanish HatEval dataset also contains instances annotated for hate speech against migrants, which is not as adjacent to incel discourse as racism and misogyny. Perhaps the model is learning from the instances annotated for hate speech against migrants to identify racism, which could still be useful in incel spaces, although not to the same extent as misogyny. In addition, given that some divergence in scheme annotation will be present regardless even with similar downstream tasks, the boost in performance is proof that the model is able to generalize even when using datasets which are not completely aligned with the target task.

With relation to I-mBERT, every combination but the one only using $\texttt{HatEval}_{\texttt{en}}$ improves the performance of the model on IFS-IT. Just like for mBERT, the best results were obtained using [$\texttt{HSD}_{\texttt{fb}}$, $\texttt{HSD}_{\texttt{tw}}$], $\texttt{HSD}_{\texttt{fb}}$, and $\texttt{HSD}_{\texttt{tw}}$. In this case, the best-performing model was trained using both the Twitter and Facebook EVALITA 2018 datasets. Once again, this result suggests the annotation scheme used for the two datasets is compatible with the one we used while annotating IFS-EN and IFS-IT. When comparing the baseline and the best-performing [$\texttt{HSD}_{\texttt{fb}}$, $\texttt{HSD}_{\texttt{tw}}$] combination, we can see that the model trained using both datasets obtains a mean F_1 score of 0.721, while for the baseline the achieved mean F_1 score is 0.503. This is a very significant 21.8-point increase in performance. The standard deviation also decreases from 0.042 to 0.010 (-76.19%), suggesting that the model greatly gains in stability as well.

When comparing the overall results obtained by the mBERT and I-mBERT models, we can see that when using $\texttt{HSD}_{\texttt{fb}}$, mBERT achieved a mean F_1 score of 0.694, while I-mBERT achieved a 0.704, for a 1-point difference in performance. Using [$\texttt{HSD}_{\texttt{fb}}$, $\texttt{HSD}_{\texttt{tw}}$], mBERT achieved a mean F_1 score of 0.690, while I-mBERT achieved a 0.721, for a 3-point difference in performance. As such, we can conclude that, when combining the MLM pre-training and the dataset merging approaches, the latter only provides a marginal boost in performance. Indeed, when only training on $\texttt{IFS-EN}_{\texttt{tr}}$, the difference in performance between mBERT and I-mBERT is of 17 points, while the difference is only 2.7 points if we compare the two models after training them on the best dataset combination, i.e., [$\texttt{HSD}_{\texttt{fb}}$, $\texttt{HSD}_{\texttt{tw}}$].

It is also interesting to notice that the improvement from the baseline to the best-performing dataset combination is higher for mBERT (36.1), than for I-mBERT (21.8). This is most likely due to the fact that the margin for improvement for the baseline mBERT model is much larger, while for I-mBERT the MLM pre-training approach is already providing a substantial boost in performance, diminishing the effect of the dataset merging strategy.

The fact that using different datasets improves the performance of the multilingual models could be seen as contradicting the conclusions drawn in Chapter 4, since it could hint at the fact that incel language is not so different from general language, after all. However, since this a cross-lingual zero-shot setting, the comparison between the lexicon contained in the training set and test set cannot be made directly. Conversely, in the monolingual scenario making such a connection between the presence of specific incel lexicon and the performance of the model makes sense, because the language of all datasets is the same.

Since in the monolingual English scenario merging datasets does not improve the performance in most cases (and when it does, the improvement is not significant), future work could be conducted by repeating the same experiments only using Italian-language data for training, development, and testing. The study could also be expanded, considering few-shot and well-resourced training settings, rather than just the zero-shot setting approached in this study. This would help shed light in a more rigorous way on whether the language used in *Il forum dei brutti* is also substantially different from general Italian, just like in the case of *Incels.is* for English.

5.4 Racism and Misogyny Detection

This section discusses the detection of racism and misogyny, which represents the multi-label text classification setting of the study. Here, classification is done via two binary labels, i.e., each instance is annotated with a "racist" label, which can either be 0 or 1, and a "misogynous" label, which can also either be 0 or 1. As such, in this multi-label setting a post can be:

- neither racist nor misogynous,
- racist, but not misogynous,
- misogynous, but not racist,
- both racist and misogynous.

This annotation strategy follows the binary relevance approach used by Zhang et al. (2018), whereby two binary classification models are combined. This allows us to study the two classifications separately and predict classes that are not mutually exclusive. We adopt this methodology based on the findings of Muti et al. (2022a), as they demonstrate how treating the classes separately increases performance when predicting classes of posts annotated as "misogynous", "misogynous-aggressive", or "none".

The task is approached in mono- and cross-lingual scenarios. In the monolingual scenario, we use I-BERT, since it was the model which obtained the highest scores in the monolingual binary hate speech detection task. In this case, we train and evaluate it on the "misogynous" and "racist" annotations of `IFS-EN`. In the cross-lingual scenario, we use I-mBERT, since it obtained the highest scores in the cross-lingual scenario of the binary hate speech detection task. We train it using `IFS-EN`'s training and development partitions, and we evaluate it in a zero-shot setup on `IFS-IT`, similarly to the binary classification task.

In both the mono- and cross-lingual scenarios, the models are trained with the same text pre-processing and training parameters used in Section 5.3. Each model is trained five times, initializing them with a different seed each run and logging their metrics. In the monolingual scenario, we use BERT_{base} as our baseline, while in the cross-lingual one we use mBERT_{base}. We do this to observe the impact of MLM pre-training in both scenarios.

Since the individual labels for this setting are binary, the performance of the models is evaluated using the same metrics used for the binary hate speech detection task, i.e., precision, recall, and F_1-measure (see Section 5.3).

5.4.1 Multi-Label Classification Results

The results for the experiments discussed in this section can be found in Table 5.9, which reports validation and test F_1-measure, recall and precision at the best-performing epoch (e) over five training runs for each model, along with the standard deviation of the F_1 scores. In both the mono- and cross-lingual scenarios, only `IFS-EN` is used for training and development. In the monolingual scenario, testing is done on $\texttt{IFS-EN}_{\texttt{te}}$, while in the cross-lingual scenario we evaluate on `IFS-IT`.

In the monolingual scenario, the misogyny-detection performance obtained by BERT_{base} and I-BERT is essentially identical in terms of F_1 score,

Table 5.9: Results for the monolingual (cross-lingual) misogyny (M) and racism (R) classification setting for BERT_{base} (mBERT_{base}) and I-BERT(I-mBERT).

		Model	**(e)**	**Validation (English)**			**Test (English)**		
				F_1	**Rec**	**Prec**	**F_1**	**Rec**	**Prec**
Monoling.	M	BERT_{base}	(2)	0.759±0.009	0.737	0.783	0.804±0.014	0.800	0.808
		I-BERT	(1)	0.786±0.005	0.786	0.786	0.803±0.005	0.826	0.782
	R	BERT_{base}	(4)	0.831±0.006	0.874	0.791	0.796±0.012	0.838	0.759
		I-BERT	(1)	0.854±0.012	0.838	0.872	0.821±0.012	0.818	0.823
							Test (Italian)		
Cross-ling.	M	mBERT_{base}	(4)	0.764±0.022	0.749	0.781	0.214±0.102	0.127	0.813
		I-mBERT	(4)	0.773±0.008	0.757	0.790	0.552±0.049	0.404	0.886
	R	mBERT_{base}	(2)	0.818±0.010	0.859	0.781	0.393±0.015	0.354	0.459
		I-mBERT	(4)	0.828±0.007	0.876	0.786	0.577±0.045	0.523	0.644

the two respectively achieving a score of 0.804 and 0.803. BERT_{base}'s result is achieved on the second epoch, with very similar recall and precision, while I-BERT achieves its result on the first epoch, with higher recall compared to its precision, meaning that it is probably overfitting on successive epochs. It is interesting to notice that, despite the result being achieved on the first epoch, the standard deviation is lower than BERT_{base}'s, hinting at a more consistent range of results with less training effort. The lack of a performance boost between I-BERT and BERT_{base} is surprising, considering this setting is carried out with data from *Incels.is*, which is supposed to have very characteristic misogynous language. This could be due to the fact that the misogynous language is easily learnt by the BERT_{base} model even without MLM training; however, this seems to contradict the fact that the misogyny setting appears to be more challenging, compared to the hate speech detection task. Perhaps a higher sample number is needed to be sure of this result, given the high standard deviation of the BERT_{base} model.

With regard to the monolingual racism detection task, I-BERT (0.821 F_1) performs slightly better than BERT_{base} (0.796 F_1), with a performance boost of 2.5 points. Once again, I-BERT achieves its highest F_1 score on the first epoch, suggesting that the model is overfitting when trained for two or more epochs. In this case, however, the two models have the same standard deviation, compared to the misogyny detection task, where I-BERT had a lower standard deviation than BERT_{base}. Just like in the hate speech detection setting, the performance boost obtained by I-BERT could be due

to the model already being familiar with the novel racist language used in *Incels.is*. Once again, in order to verify whether these results can be fully trusted, a higher sample number could be beneficial, both to verify the more challenging nature of the misogyny and racism detection tasks, and to corroborate whether I-BERT actually obtains a performance boost over BERT_{base} through MLM training.

In the cross-lingual scenario, both in the misogyny and racism detection tasks, the performance of I-mBERT is far higher compared to BERT_{base}'s. As far as misogyny is concerned, I-mBERT outperforms the baseline BERT_{base} model by 33.8 points, while in the racism detection task it outperforms the baseline by 18.4 points. These results seem to be a very strong hint at the fact that MLM training on the domain of the target language can greatly increase the performance of a model even without any downstream task training in said target language. If this method were to be applicable to other domains, this could represent a significant result for low-resource languages, as performance could be greatly improved in an unsupervised way for many classification tasks even without any available supervised datasets in the low-resource target language.

5.5 Hate Speech Forecasting

The third setting of this study involves the forecasting of hate speech. In the context of an Internet forum, we define forecasting as the capability of predicting how many posts will contain hateful content following an original post p' as soon as it has been posted. We conceptualize the amount of hate generated in a thread as the ratio between the number of hateful posts following p' and the total number of posts contained in the thread it has started. Based on this rationale, we build two datasets in which each p' is paired to a *hate score* in the range $[0, 100]$, indicating how much hate it has generated, with the extremes representing that none or all of the thread's posts are considered hateful.

We address the forecasting setting as a regression problem and use said datasets to train I-BERT and I-mBERT to output continuous $[0, 100]$ hate scores. We do this by adding a 1D linear output layer on top of them. We split both hate score datasets into training, development, and test partitions with ratios of 70/15/15 and use them to train and evaluate monolingual and multilingual models. Following the approach of Kang et al. (2018), our

Table 5.10: Statistics of the predicted labels on IFU-22-EN and IFU-22-IT, showing the number of posts labeled by the models as being hate speech.

Dataset	HS	%	No HS	%
IFU-22-EN	836,974	17.59	3,919,908	82.41
IFU-22-IT	282,724	44.30	355,419	55.70

Table 5.11: Sample of 10 instances from IFU-22-IT which were labeled as hateful by I-mBERT. The silver label is assigned by the model, while the gold label is assigned by a human annotator.

Text	Silver	Gold
Hai sentito cos'ha detto Gionnyred?Sta zitto e impara le regole dell'attrazione,razza di coglione.	1	0
Io ho sempre saputo che nell'Emilia Romagna ci nascono solo tori e checche	1	1
io vi odio tutte, belle o brutte, grasse o magre, il discorso si è capito benissimo	1	1
ai brutti veri le ragazze non si avvicinano neanche per un'amicizia	1	1
Bhe' a livello pratico non ha senso infatti, pero' boh magari cosi ci si vergogna di meno..	1	0
Buonasera dolce fanciulla, a nome di tutto il forum.	1	0
veramente hai detto che appartengo alla razza dei cessi schifosi	1	0
forse perchè le irriti...	1	0
No, la presa per il culo è che ti sei chiamata bruttoccia pur non considerandoti tale	1	0
dove sarebbe la mostruosità mi sfugge..	1	0

English and Italian baselines are the means of the scores contained in the development and test partitions of the obtained datasets.

5.5.1 Automatic Dataset Labeling

We refer to the datasets used in this setting as IFSS-EN (Incel Forum Score Dataset, English) and IFSS-IT (Incel Forum Dataset, Italian). The two datasets are obtained by automatically labeling posts contained in IFU-22-EN and IFU-22-IT (see Section 4.2) for hate speech. We label IFU-22-EN using I-BERT, trained only on $\texttt{IFS-EN}_{\texttt{tr}}$, while to label IFU-22-IT we use I-mBERT, trained on $[\texttt{HSD}_{\texttt{fb}}, \texttt{HSD}_{\texttt{tw}}]$.

As Table 5.10 shows, labeling IFU-22-EN resulted in 17.59% of its posts

being labeled as hate speech, while `IFU-22-IT` was assigned hate speech labels for 44.30% of its instances. Given that the percentage for `IFU-22-IT` appears to be excessively high, we manually verify a sample of 10 random instances which were labeled as hateful by I-mBERT. As shown in Table 5.11, most of the instances are mislabeled by the model, which jeopardizes the trustworthiness of `IFSS-IT` for the hate speech forecasting task.

The high percentage is also coherent with the results previously reported in Table 5.8 for I-mBERT, when tested on on `IFS-IT` (which is a subset of `IFU-22-IT`): the model has high recall (0.842), meaning that it rarely predicts false negatives, but low precision (0.630), indicating a larger number of false positives. In other words, the model tends to be too strict and is too eager to flag posts as containing hate speech.

Further evidence is provided by Figure 5.2, which shows histograms for the number of hateful posts in the two forums. `IFU-22-EN`'s distribution is clearly shifted to the left, while the one for `IFU-22-IT` is much closer to a normal distribution, in which the model seems to guess at random. As such, only the results for the monolingual task can be considered fully trustworthy, while the results for the multilingual automatic labeling process do not seem entirely reliable. Still, for the sake of testing whether the used model can still learn even from data of poorer quality, we report the results for the Italian dataset as well.

5.5.2 Hate Score Dataset

We use these binary decisions to compute a silver hate score for each OP in the two datasets. We define the sum H of the number of hateful posts in a thread t_j as:

$$H(t_j) = \sum_{i=0}^{N} p^h_{j,i} \tag{5.1}$$

where $p^h_{j,i}$ is a binary prediction assigned by a model, indicating whether the post $p_{j,i}$ is hateful ($p^h_{j,i} = 1$) or not ($p^h_{j,i} = 0$), with $i \in \{1, \ldots, N\}$ and N indicating the total number of posts in a thread.

The hate score S_j of a thread t_j is then calculated as:

$$S_j = 100\frac{H(t_j)}{N} \tag{5.2}$$

Table 5.12: Distribution statistics of the hate score datasets, showing the quartiles, mean and standard deviation for the ratios attributed to each OP text. The overflow rows show the statistics for the scores when excluding 0 values.

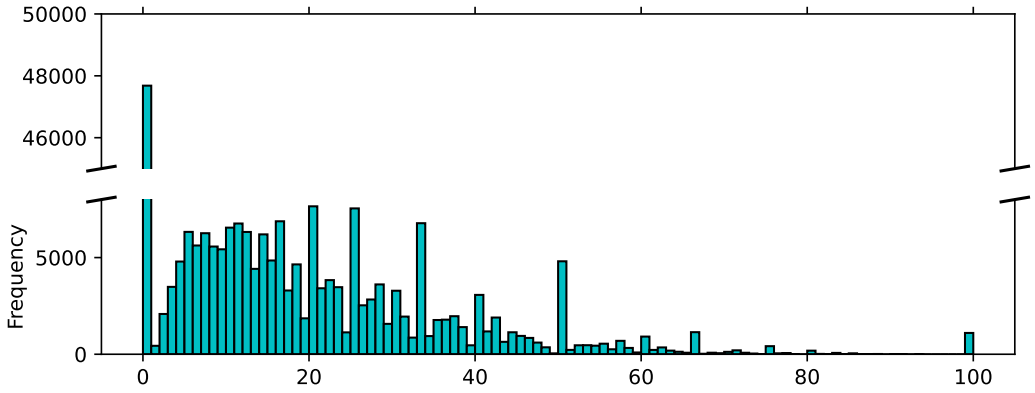
Dataset	**Mean**	**25%**	**Median**	**75%**
`IFSS-EN`	17.64 ± 17.05	04.35	13.89	26.32
$\texttt{IFSS-EN}_{\text{overflow}}$	22.44 ± 16.19	10.26	18.18	30.43
`IFSS-IT`	42.80 ± 21.31	30.00	42.86	55.56
$\texttt{IFSS-IT}_{\text{overflow}}$	45.74 ± 18.73	33.33	44.83	57.14

The ratio is multiplied by 100 to increase the range and sensitivity of the predictions, since calculating the MSE for near-zero values later would yield values which are all very close to zero. Therefore, $0 \leq S_j \leq 100$, with the extremes representing that none or all of the thread's posts are considered hateful.

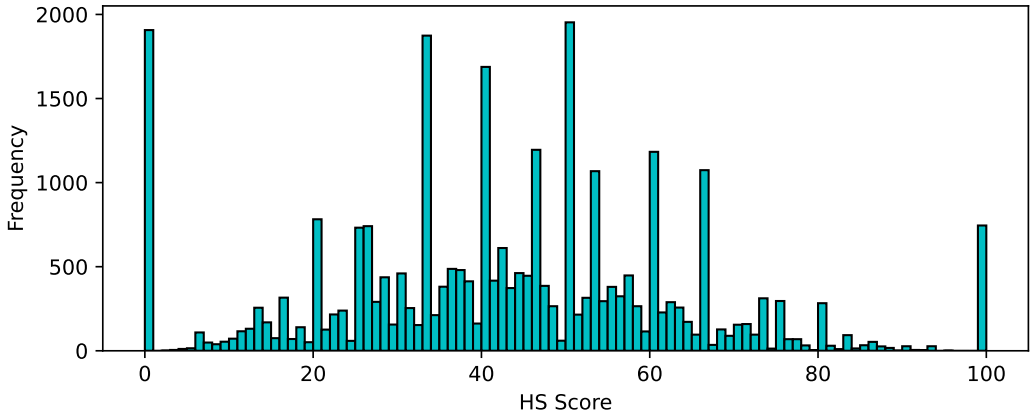
For each dataset, the hate score S_j is then assigned to the OP p'_j of the thread t_j for every $j \in \{1, \ldots, M\}$, with M indicating the total number of threads in the dataset. The resulting dataset is therefore made up of pairs (p'_j, S_j), with $j \in \{1, \ldots, M\}$. The resulting collection of (p'_j, S_j) pairs in English includes $M_{EN} = 222,994$ instances, while for the Italian $M_{IT} = 29,646$.

The statistics of the hate score datasets are listed in Table 5.12, while Figure 5.2 shows histograms for the distribution of the scores. The distribution of the English corpus is clearly skewed to the left, with a median of 13.89, indicating that most original posts tend to trigger a small amount of hateful responses. Conversely, the Italian distribution resembles a Gaussian with a median of 42.86, except for the outliers at the two extremes. This reflects a much wider and uniform range in the amount of hate predicted by I-mBERT for the posts extracted from *Il forum dei brutti*.

It is clear that many of the original posts in both forums trigger no hate, while a smaller number trigger a plethora of hateful responses. The number of completely non-hateful threads is much higher in the English OP–score dataset while, comparatively, the number is much lower in the Italian one, where it is on par with the center of the distribution. As regards the number of threads with a hate score of 100, the opposite is true: *Il forum dei brutti* has a much higher percentage, which is due to the fact that in most of its



(a) IFSS-EN



(b) IFSS-IT

Figure 5.2: Histograms for the two hate score datasets, including 0 values.

threads which only have one reply the only response is hateful (515 out of the 921 threads with a single reply).

Figure 5.2 also allows us to notice an interesting pattern in IFSS-EN's histogram, by which two curves can be traced atop the histogram bars. The first curve follows the majority of the histogram intervals, while the second appears to follow the same trend, but only for certain intervals. In the second shape, the frequency of the included hate scores is also much higher. This is most likely because we are producing this distribution by dividing integer numbers, which means certain combinations of divided numbers will be much more frequent than others, ultimately producing a second distribution

Table 5.13: Performance in terms of MSE and MAE for the forecasting setting, for the monolingual and cross-lingual scenarios (e=training epoch, b=baseline).

e	Monolingual				Cross-lingual			
	Validation		Test		Validation		Test	
	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
1	**188.63**	10.01	**181.19**	9.95	590.98	19.11	586.65	19.37
2	192.71	10.41	186.28	10.36	466.27	16.60	462.58	16.71
3	195.50	**10.00**	188.51	**9.94**	436.57	16.05	432.68	16.12
4	203.52	10.29	196.25	10.24	**425.13**	**15.82**	**421.70**	**15.95**
b	296.18	13.28	286.44	13.17	461.84	16.55	457.47	16.56

which replicates the main distribution, albeit with far higher frequencies. The reason this phenomenon cannot be observed in `IFSS-IT` is that the labels assigned to the posts produce a distribution which is much more uniform, thus not allowing for the same pattern to emerge.

5.5.3 Hate Speech Forecasting Results

Table 5.13 shows the results, recorded over four epochs. We set the maximum number of epochs at four because in the cross-lingual scenario the tuning converges on the fourth epoch.

Monolingual scenario I-BERT performs better than the baseline right from the first epoch, achieving its top performance with an MSE of 181.19 on the test set, 36.74% lower than the baseline. This indicates that the model is reasonably effective at forecasting the amount of hate that an original post is going to generate. As regards the mean absolute error (MAE), the model obtains its top performance on the third epoch, with a MAE of 9.94, compared to the 13.17 of the baseline. The fact that both metrics worsen on the fourth epoch on both the validation and test sets is a strong hint at the fact that the model is most likely overfitting on successive epochs.

Cross-lingual scenario The forecasting capabilities of I-mBERT are not as good, with the best MSE on the Italian test set being 421.70, which corresponds to a MAE of 15.95. The performance gap from the baseline is

also not as significant as in the monolingual scenario, with a delta of only 7.82% in terms of MSE. In addition to the difficulty added by the cross-lingual component, the noisier silver data produced by a lower-performing single-post classification model makes effective forecasting more challenging, which is also reflected by the slow convergence after additional epochs.

These results, particularly those in the monolingual setting, suggest that it should be possible to estimate the amount of hate that a post is likely to trigger —just by looking at its textual content— as soon as it has been posted, although the prediction quality has room for improvement. Especially as regards the cross-lingual scenario, it is possible that using a better multilingual hate speech classification model could allow us to produce a better Italian hate score dataset, which would allow us to gather more reliable regression results.

5.6 Experimental Contributions

We introduce two novel supervised datasets, `IFS-EN` and `IFS-IT`, obtained by annotating posts sampled from `IFU-22-EN` and `IFU-22-IT`, respectively. The datasets come with two binary annotation labels, "misogynous" and "racist", which are then used to assign the "hs" hate speech label. These two datasets can be used to train models for the detection of hate speech, racism, and misogyny, specifically in the context of incel forums. As we have shown, they can also be used to train models for the hate speech forecasting task, by labeling them in an unsupervised fashion.

With relation to the hate speech detection setting, i.e., the binary classification task, we contribute the following in mono- and cross-lingual scenarios:

1. We carry out an analysis of the effects of MLM pre-training on the classification performance of different Transformer models.

2. We test the effectiveness of dataset merging as a strategy for improving the performance of pre-trained Transformer models.

3. We combine both strategies to find an optimal final model configuration for the hate speech detection task.

As far as the second setting is concerned, i.e., the multi-label classification task, our contribution consists in applying different models to the misogyny

and racism detection task, in mono- and cross-lingual scenarios. The models include: BERT_{base} and BERT models developed through MLM training for the hate speech detection setting.

As regards the regression setting, we contribute the following:

1. We propose a novel, simple method for the forecasting of hate speech in online forums.

2. In both mono- and cross-lingual scenarios, we test the effectiveness of MLM-enhanced versions of BERT_{base} and mBERT_{base} for the regression task of predicting how much hate speech the first post of a forum thread is going to generate.

Chapter 6

Conclusions and Future Work

In this thesis, we have attempted to answer three research questions: (*i*) Can we build novel resources that facilitate the automatic identification of hate speech in the niche context of incel forums, in both monolingual and, especially, cross-lingual scenarios? (*ii*) Can we adapt and optimize existing automatic hate speech identification systems to effectively identify incel hate speech? (*iii*) Can we forecast whether a post will generate hate by inciting other users to respond in a hateful manner? We were spurred to investigate these questions by the necessity to sanitize and moderate social media platforms, focusing in particular on incel Internet forums.

To answer the first question, we produced two unsupervised datasets, obtained by crawling two incel forums, *Incels.is* (in English) and *Il forum dei brutti* (in Italian). We scraped their contents and annotated subsets of them with independent binary labels for "misogyny" and "racism", which can be combined to produce binary "hate speech" labels. Doing so, we have obtained datasets which can be used to train models for the identification of hate speech, misogyny, and racism, specifically within the linguistic domain of inceldom. We release all of these resources publicly, as part of our contribution to the research community.[1]

Approaching the second question entailed using the resources we have produced in synergy with existing Transformer models. In our experiments, we enhanced the performance of the models we employed by using a series of strategies, both in monolingual (English) and cross-lingual (English and Italian) scenarios. We improved their performance by adopting: (*i*) an unsuper-

[1] All datasets available at: `https://zenodo.org/record/7879341`

vised approach, by which we pre-trained models using the masked language modeling (MLM) task and (*ii*) a supervised approach, using various monolingual and multilingual combinations of datasets for downstream fine-tuning on the hate speech identification task. We find that pre-training BERT and mBERT on the MLM task using training data from the two forums increased the performance of the models in almost all settings. The results are especially promising in the cross-lingual scenario, for which we obtain a 17-point absolute F_1-measure improvement in the binary task, and a 34-point and 18-point increase in the misogyny and racism identification tasks, respectively. The dataset merging strategy was also successful, once again especially in the zero-shot cross-lingual scenario: after MLM pre-training, our best mBERT model obtained a 22-point performance boost using a combination of our English supervised dataset and two existing datasets in Italian. These improvements are significant not just for this second research question, but also for the third one, as they provide us with better models in terms of producing more accurate hate speech predictions and, therefore, more accurate hate scores with which to train our regression models.

Finally, we attempted to answer the third research question by proposing a novel, simple method of conceptualizing the forecasting of hate speech in online forums. Defining the potential of a post to produce hate as the share of hateful replies it obtains over the total of all its replies, we automatically labeled our unsupervised datasets for hate speech and built "hate score" datasets, in which each first post of a thread is assigned the ratio of hateful posts within its thread. This provided us with two datasets with 0-100 scores, one in English and one in Italian, which we have used to train BERT and mBERT MLM-enhanced models for regression. We find that while the task is still very challenging in the cross-lingual scenario, the monolingual scenario is more promising, with our model beating the MSE baseline by 37%.

In future work, based on Pelicon et al. (2021), we plan to expand the range of languages with regard to the datasets used for data augmentation. With the aim of combining existing data with our novel resources, German-language datasets annotated for hate speech (e.g., Modha et al. (2019)) represent one of the most prominent candidates for further experiments, due to the similarity between English and German. The same concept can be applied to our Italian dataset, which we could augment with additional Spanish resources (Fersini et al., 2018) and other datasets compiled in romance languages (e.g., Portuguese: Fortuna et al. (2019)). With relation to the cross-lingual scenario, it would be interesting to verify whether training only

on combinations of Italian-language datasets yields better results than also using $\texttt{IFS-EN}_{\texttt{tr}}$, when testing on IFS-IT. Experimenting with different sub-samples of these datasets could also be a viable strategy, as we have shown how in some cases sub-sampling can be more effective than using all of the available training data.

With the resources we have already produced, the performance of the models could also be improved by using not just the textual content of the posts, but also the content of the post a user is replying to, along with the whole textual context of the thread. This would most likely allow models to better discern whether a post can be considered hateful or not, directly improving the performance on the identification task, but also the forecasting task, as the automatic labeling of the threads would be more accurate, producing better hate score datasets.

With relation to the modern diachronic study conducted in Chapter 4, further research could be carried out to verify the time needed for the language of inceldom to change enough to warrant updating training resources. For example, this problem could be approched by extracting samples from the forums at equal intervals over their lifespans, annotating them for hate speech, and evaluating binary classifiers on these supervised datasets.

Another point that needs to investigated further is that, in the modern diachronic study, our keyword method is intrinsically not capable of detecting all terms which are used with offensive novel meanings. Further research could be carried out to find a way to identify terms which are normally innocuous, but that are used in a hateful way depending on the context of the sentence or thread, or even based on the forum as a whole.

In addition, as anticipated in Section 4.1, another aspect that needs addressing is the fact that the content of the datasets is specific to a niche of misogyny and racism. Although some compatibility has been observed in the cross-lingual scenarios in this thesis, further work could be done to verify whether models trained on the compiled datasets can generalize to general misogynous and racist language on broader social media platforms.

As regards model architectures, we have observed that RoBERTa models also improve on the monolingual binary classification task after undergoing MLM training, albeit more slowly than BERT models. Therefore, it would be interesting to verify, given enough computational power, time, and training data, whether the performance of RoBERTa models can also be competitive, along with other architectures we have not explored.

Since social media platforms are multi-modal content spaces, a partic-

ularly interesting option is also represented by multi-modal models. Many models for multi-modal identification are available out-of-the-box, and could be applied with relatively little effort, provided that existing datasets can be effectively integrated with our resources for the task at hand. Building upon research on multi-modal misogyny identification (e.g., Muti et al. (2022c)), we could for example leverage information not just from textual content, but also the images and videos included in the body of a post.

Cross-domain hate speech classification is another possible avenue of research. In preliminary experiments, we tested the generalizability of our models on the Contextual Abuse Dataset (Vidgen et al., 2021), but its mismatching annotation scheme lead to unsatisfactory results. This was the only available thread dataset which was relatively close to our purposes; however, it is annotated for types of abusive language, which are not suitable labels for the tasks we are approaching. The creation of other datasets annotated both at a thread and post level, not necessarily in the domain of inceldom, could greatly benefit research both with regard to the classification and forecasting of hate speech.

A self-learning process could also be used to improve the models, as described in Jurkiewicz et al. (2020), having them predict silver labels in order to produce training data which can be re-used for training. However, long training times could be an issue, as using this algorithm involves labeling a chunk of instances and then re-training from scratch for a number of iterations.

Finally, the forecasting of hate speech is arguably one of the areas in which more opportunities for further research can be found. As the approach we used in this study was rather naive and simple, we plan to approach this task by implementing more sophisticated methods. We could for example extract features for shift in sentiment and topic flow, like in Almerekhi et al. (2020), and implement temporal and propagation features in our experiments, following the approaches of Lin et al. (2021) and Meng et al. (2023).

This work has been accepted at RANLP 2023 in the form of a short paper and is set to appear in the proceedings of the conference in September 2023.

Bibliography

Hind Almerekhi, Haewoon Kwak, Joni Salminen, and Bernard J. Jansen. Are these comments triggering? predicting triggers of toxicity in online discussions. In *Proceedings of The Web Conference 2020*, WWW '20, page 3033–3040, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450370233. doi: 10.1145/3366423.3380074. URL https://doi.org/10.1145/3366423.3380074.

Sai Saketh Aluru, Binny Mathew, Punyajoy Saha, and Animesh Mukherjee. Deep Learning Models for Multilingual Hate Speech Detection, December 2020. URL http://arxiv.org/abs/2004.06465. arXiv:2004.06465 [cs].

Paul Baker. Corpus methods in linguistics. *Research methods in linguistics*, 93, 2010.

Marco Baroni and Silvia Bernardini. Bootcat: Bootstrapping corpora and terms from the web. In *LREC*, pages 1313–1316, 2004.

Valerio Basile, Cristina Bosco, Elisabetta Fersini, Debora Nozza, Viviana Patti, Francisco Manuel Rangel Pardo, Paolo Rosso, and Manuela Sanguinetti. 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*, pages 54–63, 2019.

Valerio Basile, Di Maro Maria, Croce Danilo, Lucia C Passaro, et al. Evalita 2020: Overview of the 7th evaluation campaign of natural language processing and sspeech tools for italian. In *Proceedings of the Seventh Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2020)*, pages 1–7. CEUR-ws, 2020.

Victoria Bobicev and Marina Sokolova. Inter-annotator agreement in sentiment analysis: Machine learning perspective. In *International Conference Recent Advances in Natural Language Processing*, pages 97–102, 2017.

Cristina Bosco, Felice Dell'Orletta, Fabio Poletto, Manuela Sanguinetti, and Maurizio Tesconi. Overview of the EVALITA 2018 Hate Speech Detection Task. In Tommaso Caselli, Nicole Novielli, Viviana Patti, and Paolo Rosso, editors, *EVALITA Evaluation of NLP and Speech Tools for Italian*, pages 67–74. Accademia University Press, 2018. ISBN 978-88-319-7842-2 978-88-319-7869-9. doi: 10.4000/books.aaccademia.4503. URL `http://books.openedition.org/aaccademia/4503`.

Tommaso Caselli, Valerio Basile, Jelena Mitrović, and Michael Granitzer. HateBERT: Retraining BERT for Abusive Language Detection in English, February 2021. URL `http://arxiv.org/abs/2010.12472`.

François Chollet. *Deep learning with Python*. Manning, Shelter Island, 2018. ISBN 978-1-61729-443-3.

Eugenio Coseriu. Los conceptos de dialecto, nivel y estilo de lengua y el sentido propio de la dialectología. *LEA: Lingüística española actual*, 3(1): 1–32, 1981.

Snehil Dahiya, Shalini Sharma, Dhruv Sahnan, Vasu Goel, Emilie Chouzenoux, Víctor Elvira, Angshul Majumdar, Anil Bandhakavi, and Tanmoy Chakraborty. Would your tweet invoke hate on the fly? forecasting hate intensity of reply threads on twitter. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, KDD '21, page 2732–2742, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450383325. doi: 10.1145/3447548.3467150. URL `https://doi.org/10.1145/3447548.3467150`.

Shanty Das. Inside the violent, misogynistic world of tiktok's new star, andrew tate. `https://www.theguardian.com/technology/2022/aug/06/andrew-tate-violent-misogynistic-world-of-tiktok-new-star`, 2022. [Online; accessed 9-June-2023].

Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. Automated hate speech detection and the problem of offensive language.

In *Proceedings of the International AAAI Conference on Web and Social Media*. arXiv, May 2017. doi: 10.48550/ARXIV.1703.04009. URL https://arxiv.org/abs/1703.04009.

Ona De Gibert, Naiara Perez, Aitor García-Pablos, and Montse Cuadros. Hate speech dataset from a white supremacy forum. *arXiv preprint arXiv:1809.04444*, 2018.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, May 2019. URL http://arxiv.org/abs/1810.04805.

Pierce Alexander Dignam and Deana A. Rohlinger. Misogynistic Men Online: How the Red Pill Helped Elect Trump. *Signs: Journal of Women in Culture and Society*, 44(3):589–612, March 2019. ISSN 0097-9740, 1545-6943. doi: 10.1086/701155. URL https://www.journals.uchicago.edu/doi/10.1086/701155.

Tracie Farrell, Miriam Fernandez, Jakub Novotny, and Harith Alani. Exploring Misogyny across the Manosphere in Reddit. In *Proceedings of the 10th ACM Conference on Web Science*, pages 87–96, Boston Massachusetts USA, June 2019. ACM. ISBN 978-1-4503-6202-3. doi: 10.1145/3292522.3326045. URL https://dl.acm.org/doi/10.1145/3292522.3326045.

Tracie Farrell, Oscar Araque, Miriam Fernandez, and Harith Alani. On the use of jargon and word embeddings to explore subculture within the reddit's manosphere. In *12th ACM Conference on Web Science*, WebSci '20, page 221–230, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450379892. doi: 10.1145/3394231.3397912. URL https://doi.org/10.1145/3394231.3397912.

Elisabetta Fersini, Paolo Rosso, and Maria Anzovino. Overview of the task on automatic misogyny identification at ibereval 2018. 2150:214–228, 2018. URL https://ceur-ws.org/Vol-2150/overview-AMI.pdf.

Elisabetta Fersini, Debora Nozza, and Paolo Rosso. Ami @ evalita2020: Automatic misogyny identification. In Valerio Basile, Danilo Croce, Maria Di Maro, and Lucia C. Passaro, editors, *Proceedings of the 7th evaluation campaign of Natural Language Processing and Speech tools for Italian (EVALITA 2020)*, Online, 2020. CEUR.org.

Paula Fortuna, João Rocha da Silva, Juan Soler-Company, Leo Wanner, and Sérgio Nunes. A hierarchically-labeled Portuguese hate speech dataset. In *Proceedings of the Third Workshop on Abusive Language Online*, pages 94–104, Florence, Italy, August 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-3510. URL https://aclanthology.org/W19-3510.

Hadley Freeman. Tradwives: The new trend for submissive women has a dark heart and history. https://www.theguardian.com/fashion/2020/jan/27/tradwives-new-trend-submissive-women-dark-heart-history, 2020. [Online; accessed 9-June-2023].

Debbie Ging. Alphas, Betas, and Incels: Theorizing the Masculinities of the Manosphere. *Men and Masculinities*, 22(4):638–657, October 2019. ISSN 1097-184X, 1552-6828. doi: 10.1177/1097184X17706401. URL http://journals.sagepub.com/doi/10.1177/1097184X17706401.

Debbie Ging and Eugenia Siapera. Special issue on online misogyny. *Feminist Media Studies*, 18(4):515–524, 2018. doi: 10.1080/14680777.2018.1447345. URL https://doi.org/10.1080/14680777.2018.1447345.

Alyssa M. Glace, Tessa L. Dover, and Judith G. Zatkin. Taking the black pill: An empirical analysis of the "Incel". *Psychology of Men & Masculinities*, 22(2):288–297, April 2021. ISSN 1939-151X, 1524-9220. doi: 10.1037/men0000328. URL http://doi.apa.org/getdoi.cfm?doi=10.1037/men0000328.

Omkar Gokhale, Aditya Kane, Shantanu Patankar, Tanmay Chavan, and Raviraj Joshi. Spread Love Not Hate: Undermining the Importance of Hateful Pre-training for Hate Speech Detection. December 2022. URL http://arxiv.org/abs/2210.04267. arXiv:2210.04267 [cs].

Kelly C Gothard. Exploring Incel Language and Subreddit Activity on Reddit. 2020.

Dan Hendrycks and Kevin Gimpel. Gaussian Error Linear Units (GELUs), July 2020. URL http://arxiv.org/abs/1606.08415. arXiv:1606.08415 [cs].

Bruce Hoffman, Jacob Ware, and Ezra Shapiro. Assessing the Threat of Incel Violence. *Studies in Conflict & Terrorism*, 43(7):565–587,

July 2020a. ISSN 1057-610X, 1521-0731. doi: 10.1080/1057610X.2020.1751459. URL https://www.tandfonline.com/doi/full/10.1080/1057610X.2020.1751459.

Bruce Hoffman, Jacob Ware, and Ezra Shapiro. Assessing the Threat of Incel Violence. *Studies in Conflict & Terrorism*, 43(7):565–587, July 2020b. ISSN 1057-610X, 1521-0731. doi: 10.1080/1057610X.2020.1751459. URL https://www.tandfonline.com/doi/full/10.1080/1057610X.2020.1751459.

Peter Holtz, Nicole Kronberger, and Wolfgang Wagner. Analyzing internet forums. *Journal of Media Psychology*, 2012.

Hidenori Ide and Takio Kurita. Improvement of learning for CNN with ReLU activation by sparse regularization. In *2017 International Joint Conference on Neural Networks (IJCNN)*, pages 2684–2691, Anchorage, AK, USA, May 2017. IEEE. ISBN 978-1-5090-6182-2. doi: 10.1109/IJCNN.2017.7966185. URL http://ieeexplore.ieee.org/document/7966185/.

Sylvia Jaki, Tom De Smedt, Maja Gwóźdź, Rudresh Panchal, Alexander Rossa, and Guy De Pauw. Online hatred of women in the Incels.me forum: Linguistic analysis and automatic detection. *Journal of Language Aggression and Conflict*, 7(2):240–268, November 2019. ISSN 2213-1272, 2213-1280. doi: 10.1075/jlac.00026.jak. URL http://www.jbe-platform.com/content/journals/10.1075/jlac.00026.jak.

Miloš Jakubíček, Adam Kilgarriff, Vojtěch Kovář, Pavel Rychlý, and Vít Suchomel. The tenten corpus family. *7th International Corpus Linguistics Conference CL 2013*, 07 2013.

Olga Jurasz and Kim Barker. Online misogyny: A challenge for digital feminism? *Journal of International Affairs*, 72(2):95–114, 2019. URL https://oro.open.ac.uk/66200/.

Dawid Jurkiewicz, Łukasz Borchmann, Izabela Kosmala, and Filip Graliński. ApplicaAI at SemEval-2020 Task 11: On RoBERTa-CRF, Span CLS and Whether Self-Training Helps Them, September 2020. URL http://arxiv.org/abs/2005.07934. arXiv:2005.07934 [cs].

Dongyeop Kang, Waleed Ammar, Bhavana Dalvi, Madeleine Zuylen, Sebastian Kohlmeier, Eduard Hovy, and Roy Schwartz. A Dataset of Peer

Reviews (PeerRead): Collection, Insights and NLP Applications. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1647–1661, New Orleans, Louisiana, 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1149. URL http://aclweb.org/anthology/N18-1149.

Adam Kilgarriff. Simple maths for keywords. In *Proc. Corpus Linguistics*, 2009.

Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization, January 2017. URL http://arxiv.org/abs/1412.6980. arXiv:1412.6980 [cs].

Hobson Lane, Cole Howard, and Hannes Hapke. *Natural Language Processing in Action*. Manning Publications, 2019.

Ed Latimore. The 4 unfakeable traits of a high value man. https://edlatimore.com/high-value-man/, n.d. [Online; accessed 9-June-2023].

Alyssa Lees, Jeffrey Sorensen, and Ian Kivlichan. Jigsaw @ AMI and HaSpeeDe2: Fine-Tuning a Pre-Trained Comment-Domain BERT Model. In Valerio Basile, Danilo Croce, Maria Maro, and Lucia C. Passaro, editors, *EVALITA Evaluation of NLP and Speech Tools for Italian - December 17th, 2020*, pages 40–47. Accademia University Press, 2020. ISBN 9791280136329. doi: 10.4000/books.aaccademia.6789. URL http://books.openedition.org/aaccademia/6789.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *CoRR*, abs/1910.13461, 2019. URL http://arxiv.org/abs/1910.13461.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages

7871–7880, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.703. URL https://aclanthology.org/2020.acl-main.703.

Lexical Computing Ltd. Statistic used in sketch engine. https://www.sketchengine.eu/documentation/statistics-used-in-sketch-engine/, 7 2015.

Ken-Yu Lin, Roy Ka-Wei Lee, Wei Gao, and Wen-Chih Peng. Early prediction of hate speech propagation. In *2021 International Conference on Data Mining Workshops (ICDMW)*, pages 967–974, 2021. doi: 10.1109/ICDMW53433.2021.00126.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692, 2019. URL http://arxiv.org/abs/1907.11692.

Agnes Lydia and Sagayaraj Francis. Adagrad—an optimizer for stochastic gradient descent. *Int. J. Inf. Comput. Sci*, 6(5):566–568, 2019.

Ariadna Matamoros-Fernández and Johan Farkas. Racism, Hate Speech, and Social Media: A Systematic Review and Critique. *Television & New Media*, 22(2):205–224, February 2021. ISSN 1527-4764, 1552-8316. doi: 10.1177/1527476420982230. URL http://journals.sagepub.com/doi/10.1177/1527476420982230.

Binny Mathew, Ritam Dutt, Pawan Goyal, and Animesh Mukherjee. Spread of hate speech in online social media. In *Proceedings of the 10th ACM conference on web science*, pages 173–182, 2019.

Binny Mathew, Punyajoy Saha, Seid Muhie Yimam, Chris Biemann, Pawan Goyal, and Animesh Mukherjee. Hatexplain: A benchmark dataset for explainable hate speech detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 14867–14875, 2021.

Tony McEnery and Andrew Wilson. Corpus linguistics. *The Oxford handbook of computational linguistics*, pages 448–463, 2003.

Qing Meng, Tharun Suresh, Roy Ka-Wei Lee, and Tanmoy Chakraborty. Predicting hate intensity of twitter conversation threads. *arXiv preprint*

arXiv:2206.08406, May 2023. URL https://doi.org/10.48550/arXiv.2206.08406. Accepted in Knowledge-Based Systems.

Sandip Modha, Thomas Mandl, Prasenjit Majumder, and Daksh Patel. Overview of the HASOC track at FIRE 2019: Hate Speech and Offensive Content Identification in Indo-European Languages. 2019.

Ioannis Mollas, Zoe Chrysopoulou, Stamatis Karlos, and Grigorios Tsoumakas. Ethos: a multi-label hate speech detection dataset. *Complex & Intelligent Systems*, pages 1–16, 2022.

Arianna Muti and Alberto Barrón-Cedeño. UniBO @ AMI: A Multi-Class Approach to Misogyny and Aggressiveness Identification on Twitter Posts Using AlBERTo. In Valerio Basile, Danilo Croce, Maria Maro, and Lucia C. Passaro, editors, *EVALITA Evaluation of NLP and Speech Tools for Italian - December 17th, 2020*, pages 29–34. Accademia University Press, 2020. ISBN 9791280136329. doi: 10.4000/books.aaccademia.6769. URL http://books.openedition.org/aaccademia/6769.

Arianna Muti, Francesco Fernicola, and Alberto Barrón-Cedeño. Misogyny and aggressiveness tend to come together and together we address them. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 4142–4148, Marseille, France, June 2022a. European Language Resources Association. URL https://aclanthology.org/2022.lrec-1.440.

Arianna Muti, Francesco Fernicola, and Alberto Barrón-Cedeño. Misogyny and Aggressiveness Tend to Come Together and Together We Address Them. 2022b.

Arianna Muti, Katerina Korre, and Alberto Barrón-Cedeño. UniBO at SemEval-2022 Task 5: A Multimodal bi-Transformer Approach to the Binary and Fine-grained Identification of Misogyny in Memes. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 663–672, Seattle, United States, 2022c. Association for Computational Linguistics. doi: 10.18653/v1/2022.semeval-1.91. URL https://aclanthology.org/2022.semeval-1.91.

Angela Nagle. *Kill all normies: Online culture wars from 4chan and Tumblr to Trump and the alt-right.* Zero Books, Winchester, Hampshire, UK, July 2017. doi: 10.5817/PC2018-3-270.

Enrique W. Neblett. Racism and health: Challenges and future directions in behavioral and psychological research. *Cultural Diversity and Ethnic Minority Psychology*, 25(1):12–20, January 2019. ISSN 1939-0106, 1099-9809. doi: 10.1037/cdp0000253. URL `http://doi.apa.org/getdoi.cfm?doi=10.1037/cdp0000253`.

Anh Nguyen, Khoa Pham, Dat Ngo, Thanh Ngo, and Lam Pham. An Analysis of State-of-the-art Activation Functions For Supervised Deep Neural Network. In *2021 International Conference on System Science and Engineering (ICSSE)*, pages 215–220, Ho Chi Minh City, Vietnam, August 2021. IEEE. ISBN 978-1-66544-848-2. doi: 10.1109/ICSSE52999.2021.9538437. URL `https://ieeexplore.ieee.org/document/9538437/`.

John T. Nockleby. Hate speech. In Leonard W. Levy, Kenneth L. Karst, et al., editors, *Encyclopedia of the American Constitution*, pages 1277–1279. Macmillan, New York, 2 edition, 2000.

Kostantinos Papadamou, Savvas Zannettou, Jeremy Blackburn, Emiliano De Cristofaro, Gianluca Stringhini, and Michael Sirivianos. Understanding the incel community on youtube. *CoRR*, abs/2001.08293, 2020. URL `https://arxiv.org/abs/2001.08293`.

Alan Partington. *Modern Diachronic Corpus-Assisted Discourse Studies: Corpora Volume 5, Number 2*. Edinburgh University Press, 2010. ISBN 9780748640607. URL `http://www.jstor.org/stable/10.3366/j.ctt1r2604`.

Andraž Pelicon, Ravi Shekhar, Blaž Škrlj, Matthew Purver, and Senja Pollak. Investigating cross-lingual training for offensive language detection. *PeerJ Computer Science*, 7:e559, June 2021. ISSN 2376-5992. doi: 10.7717/peerj-cs.559. URL `https://peerj.com/articles/cs-559`.

Björn Pelzer, Lisa Kaati, Katie Cohen, and Johan Fernquist. Toxic language in online incel communities. *SN Social Sciences*, 1(8):1–22, 2021.

Marco Polignano, Pierpaolo Basile, Marco de Gemmis, Giovanni Semeraro, and Valerio Basile. AlBERTo: Italian BERT Language Understanding Model for NLP Challenging Tasks Based on Tweets. In *Proceedings of the Sixth Italian Conference on Computational Linguistics (CLiC-it 2019)*, volume 2481. CEUR, 2019. URL `https:`

//www.scopus.com/inward/record.uri?eid=2-s2.0-85074851349&partnerID=40&md5=7abed946e06f76b3825ae5e294ffac14.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(1), jan 2020. ISSN 1532-4435.

Manoel Horta Ribeiro, Jeremy Blackburn, Barry Bradlyn, Emiliano De Cristofaro, Gianluca Stringhini, Summer Long, Stephanie Greenberg, and Savvas Zannettou. The evolution of the manosphere across the web. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 15, pages 196–207, 2021.

Dhruv Sahnan, Snehil Dahiya, Vasu Goel, Anil Bandhakavi, and Tanmoy Chakraborty. Better prevent than react: Deep stratified learning to predict hate intensity of twitter reply chains. In *2021 IEEE International Conference on Data Mining (ICDM)*, pages 549–558. IEEE, 2021.

Giuseppe Sansonetti, Fabio Gasparetti, Giuseppe D'aniello, and Alessandro Micarelli. Unreliable Users Detection in Social Media: Deep Learning Techniques for Automatic Detection. *IEEE Access*, 8:213154–213167, 2020. ISSN 2169-3536. doi: 10.1109/ACCESS.2020.3040604. URL https://ieeexplore.ieee.org/document/9269985/.

Joshua A Segalewitz. "You Don't Understand... It's Not About Virginity":. 2020.

Leandro Silva, Mainack Mondal, Denzil Correa, Fabrício Benevenuto, and Ingmar Weber. Analyzing the targets of hate in online social media. In *Tenth international AAAI conference on web and social media*, 2016.

Vaios Stergiopoulos, Michael Vassilakopoulos, Eleni Tousidou, and Antonio Corral. An application of ANN hyper-parameters tuning in the field of Recommender Systems. 2022.

Lyman Stone. Male sexlessness is rising but not for the reasons incels claim. `https://ifstudies.org/blog/male-sexlessness-is-rising-but-not-for-the-reasons-incels-claim`, 2018. [Online; accessed 9-June-2023].

Cagri Toraman, Furkan Şahinuç, and Eyup Yilmaz. Large-scale hate speech detection with cross-domain transfer. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 2215–2225, Marseille, France, June 2022. European Language Resources Association. URL `https://aclanthology.org/2022.lrec-1.238`.

Alessia Tranchese and Lisa Sugiura. "i don't hate all women, just those stuck-up bitches": How incels and mainstream pornography speak the same extreme language of misogyny. *Violence Against Women*, 27(14): 2709–2734, 2021. doi: 10.1177/1077801221996453. URL `https://doi.org/10.1177/1077801221996453`. PMID: 33750244.

Lewis Tunstall, Leandro Von Werra, and Thomas Wolf. *Natural language processing with transformers*. " O'Reilly Media, Inc.", 2022.

Matej Ulčar and Marko Robnik-Šikonja. Finest bert and crosloengual bert: Less is more in multilingual models. In *Text, Speech, and Dialogue: 23rd International Conference, TSD 2020, Brno, Czech Republic, September 8–11, 2020, Proceedings*, page 104–111, Berlin, Heidelberg, 2020. Springer-Verlag. ISBN 978-3-030-58322-4. doi: 10.1007/97-3-030-58323-1_11. URL `https://doi.org/10.1007/978-3-030-58323-1_11`.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention Is All You Need, December 2017. URL `http://arxiv.org/abs/1706.03762`. arXiv:1706.03762 [cs].

Bertie Vidgen, Dong Nguyen, Helen Margetts, Patricia Rossini, and Rebekah Tromble. Introducing CAD: the contextual abuse dataset. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2289–2303, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.182. URL `https://aclanthology.org/2021.naacl-main.182`.

Zeerak Waseem and Dirk Hovy. Hateful symbols or hateful people? predictive features for hate speech detection on twitter. In *Proceedings of the NAACL Student Research Workshop*, pages 88–93, San Diego, California, June 2016. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/N16-2013.

Zeerak Waseem, Thomas Davidson, Dana Warmsley, and Ingmar Weber. Understanding Abuse: A Typology of Abusive Language Detection Subtasks, May 2017. URL http://arxiv.org/abs/1705.09899. arXiv:1705.09899 [cs].

Ronald J. Williams and David Zipser. A learning algorithm for continually running fully recurrent neural networks. *Neural Computation*, 1(2):270–280, 1989. doi: 10.1162/neco.1989.1.2.270.

Peng Xu, Xiatian Zhu, and David A. Clifton. Multimodal Learning with Transformers: A Survey, May 2023. URL http://arxiv.org/abs/2206.06488. arXiv:2206.06488 [cs].

Olivia Young. What Role Has Social Media Played in Violence Perpetrated by Incels? 2019.

Min-Ling Zhang, Yu-Kun Li, Xu-Ying Liu, and Xin Geng. Binary relevance for multi-label learning: An overview. *Frontiers of Computer Science*, 12(2):191–202, April 2018. ISSN 2095-2228, 2095-2236. doi: 10.1007/s11704-017-7031-7. URL http://link.springer.com/10.1007/s11704-017-7031-7.

Linda Zhou, Andrew Caines, Ildiko Pete, and Alice Hutchings. Automated hate speech detection and span extraction in underground hacking and extremist forums. *Natural Language Engineering*, pages 1–28, 2022.