Paul Röttger, St Edmund Hall, University of Oxford

Improving the Evaluation and Effectiveness of Hate Speech Detection Models

Thesis submitted in partial fulfilment of the requirements for the degree of DPhil in Social Data Science at the Oxford Internet Institute, University of Oxford

Date of submission: March 23rd, 2023

Supervisors: Prof. Janet B. Pierrehumbert and Prof. Helen Margetts

Word count (excl. bibliography): 36,000 words + ca. 3,000 word-equivalents in tables and figures

ACKNOWLEDGMENTS

I am and always will be deeply grateful for the guidance and support of my supervisors, Janet Pierrehumbert and Helen Margetts. Janet taught the first NLP course I ever took, and changed so much for me when she agreed to take me on as her DPhil student. In everything I worked on, Janet pushed me to explore new directions, and to think deeply about social and linguistic factors in language modelling, rather than approaching it as just another engineering problem. Helen challenged me to have an impact outside of academia, connecting my research to industry practice and policy. She helped me navigate Oxford's institutional complexities, and encouraged me to connect and collaborate beyond my Oxford bubble.

My last three years also would not have been the same without Bertie Vidgen. Bertie was the first to introduce me to hate speech research in Oxford. From mentoring me at the beginning of my DPhil, to starting and running Rewire together, I am deeply grateful for everything he taught me along the way, as a colleague and as a friend. Thank you also to Taha Yasseri, who put me in touch with Bertie in the first place.

I feel fortunate to work in a field that is built on collaboration and co-authorship. In Oxford, Valentin Hofmann and Hannah Rose Kirk were the best academic sparring partners one could wish for, and I am very excited for all the great things they will go on to do. Thank you also to all other members of the Pierrehumbert Language Modelling Group for generously sharing ideas and giving feedback. In London, the Alan Turing Institute's Online Safety team – not just Helen, Bertie and Hannah, but also Pica Johansson and Angus Williams – generously took me in for a research visit. In Milan, Dirk Hovy was a great mentor and academic role model, and I cannot wait to do more work together. Here, too, I am grateful to the lab / aperitivo group – especially Debora Nozza, Amanda Cercas-Curry, Giuseppe Attanasio, Federicho Bianchi, Anne Lauscher and Matthias Orlikowski – for all the fun working and not-working together. Scattered across the world, I also want to thank Balazs Vedres, Zeerak Talat, Dong Nguyen, Max Bartolo, Haitham Seelawi and Janosch Haber. Thanks to the anonymous reviewers providing feedback on my work, who in their own way, too, were helpful collaborators.

My studies were mainly funded by the German Academic Scholarship Foundation. Seeing people impacted by post-Brexit tuition fees in the UK has made me particularly aware of the privilege of receiving this funding, available to me simply because of where I happened to be born. I still feel deeply grateful for it. Thank you as well for additional funding for conferences and travel from the Alan Turing Institute and Bocconi University.

Like many researchers, especially in NLP, I have also profited from the less-visible labour of data annotators, helping with my own work and enabling the work of others. I thank them. Similarly, I am thankful for the support of all admin and research staff at the Oxford Internet Institute, St Edmund Hall, the Alan Turing Institute and Bocconi University.

Lastly, a personal note of thanks to my housemates – especially Dan, Zak and Moritz in Oxford, and Bonnie and Vince in London – for the joy of our friendship and living together; to Edu, Selina, Hubert, David, Julia, Bob, Pat and Nayana for great times in Oxford and elsewhere; to my friends back in Germany for making me feel at home wherever I am coming back from; to my parents, Linda and Peter, and my brother Felix, for listening and for caring; and to Hannah L for sharing her love with me.

This thesis truly was a group effort. Thank you all!

Content Warning

This thesis is about online hate speech and models for detecting it. Therefore, I discuss hate speech and other online harms in detail. I avoid reproducing hate speech where possible, but I do at several points provide illustrative examples of hateful language. For hateful slurs and profanity, I generally replace the first vowel with an asterisk.

ABSTRACT

Online hate speech is a widespread and deeply harmful problem. To tackle hate at scale, we need models that can automatically detect it. This has motivated research in Natural Language Processing (NLP) to develop text-based hate speech detection models. In recent years, these models have improved substantially, following general advances in language modelling.

In my thesis, I show that impressive headline results paint an incomplete picture of model quality. I argue that much progress in hate speech detection so far has rested on simplifying assumptions, which, while useful in some settings, we need to move past in order to develop more truly effective models. In particular, I argue that current standards for model evaluation tend to be overly aggregated, static, monolithic and English language-centric, because of four simplifying assumptions that I use to structure my thesis.

I discuss core concepts in an introduction and literature review. Then, I present four Chapters, which each challenge one of the four simplifying assumptions. Assumption 1 is that *model accuracy equals model quality*. I introduce a suite of functional tests for hate speech detection models, which enables fine-grained diagnostic insights and reveals critical weaknesses in seemingly accurate models. Assumption 2 is that *hate speech today equals hate speech tomorrow*. I find that model performance degrades over time and explore temporal adaptation as a remedy. Assumption 3 is that *hate speech for me equals hate speech for you*. I evidence subjectivity in labelling hate speech and introduce two contrasting data annotation paradigms for managing subjectivity. Assumption 4 is that *hate speech in English equals all hate speech*. I explore data-efficient strategies for expanding detection into more under-resourced languages.

Overall, my thesis seeks to 1) work towards better, more comprehensive quality standards for hate speech detection models, and 2) improve models along these standards.

CONTENTS

1 Introduction **1**

1.1 Thesis Structure 3

2 Literature Review **7**

2.1 What Is Hate Speech? 7

2.2 How Can We Detect Hate Speech? 9

2.3 How Do We Test Hate Speech Detection Models? 12

3 HateCheck - Functional Tests for Hate Speech Detection Models **15**

3.1 Introduction 15

3.2 Constructing HateCheck 17

3.3 Testing Models with HateCheck 23

3.4 Limitations 29

3.5 Related Work 31

3.6 Conclusion 32

4 Temporal Adaptation of BERT and Performance on Downstream Document Classification: Insights from Social Media **33**

4.1 Introduction 33

4.2 Related Work 36

4.3 Experiments 37

4.4 Discussion 48

4.5 Conclusion 51

5 Two Contrasting Data Annotation Paradigms for Subjective NLP Tasks 53

5.1 Introduction 53

5.2 The Descriptive Annotation Paradigm: Encouraging Annotator Subjectivity . . 55

5.3 The Prescriptive Annotation Paradigm: Discouraging Annotator Subjectivity . 58

5.4 An Illustrative Annotation Experiment 60

5.5 Conclusion 61

6 Data-Efficient Strategies for Expanding Hate Speech Detection into Under-Resourced Languages 62

6.1 Introduction 62

6.2 Experiments 65

6.3 Discussion and Recommendations 73

6.4 Related Work 78

6.5 Conclusion 79

7 Conclusion 81

A Appendices for Chapter 3: HateCheck 85

B Appendices for Chapter 4: Temporal Adaptation 95

C Appendices for Chapter 5: Annotation Paradigms 101

D Appendices for Chapter 6: Data-Efficient Low-Resource Hate Detection 106

E Co-Authorship and Contributions 112

F Funding Statement 114

1 INTRODUCTION

The proliferation of online hate speech is one of the biggest challenges for internet governance today. Hate speech, which is most commonly defined as abusive content targeting people because of their race, gender or other protected characteristics, causes direct psychological harms to its targets and to others exposed to it (Vidgen et al., 2021a). It causes wider societal harms by ostracising people from social discourse, undermining social norms of justice and fairness (Matsuda, 1993; Waldron, 2012). In some cases, online hate even predicts offline hate crime (Williams et al., 2020; Müller and Schwarz, 2021; Lorenz-Spreen et al., 2022).

There is an enormous amount of hate speech on the internet. The relative proportion of hateful content is low, but because there is so much online content, the absolute number of hateful messages, comments and posts is staggeringly high. For example, just 0.02% of Facebook's content views in Q2 2022 showed hate speech, but over 13.5 million pieces of hate speech were taken down during that time, according to Meta's own reporting (Meta, 2022).

Human oversight and moderation alone cannot fully address the problem of online hate speech, because human review does not scale. It is time-intensive and costly. It also risks exposing moderators themselves to harm (Gillespie, 2018; Kirk et al., 2022a). Moderators often live and work in the Global South, employed by outsourcing companies that put in place minimal protections for their wellbeing (Foxglove, 2021; Perrigo, 2023), and some moderators have experienced severe mental health issues due to the content they had to review (BBC, 2020; Elliott and Parmar, 2020). These real and consequential damages caused by online hate speech, along with the shortcomings of human moderation, have motivated a growing body of research, from academia and industry, into the automated detection of hateful content – particularly in the field of natural language processing (NLP).

There has been much progress in the development of hate speech detection models over the last decade. Early models relied on simple machine learning techniques like support vector machines (e.g. Warner and Hirschberg, 2012; Tulkens et al., 2016) or logistic regression (e.g. Djuric et al., 2015; Talat and Hovy, 2016). Following general trends and advances in language

modelling, hate speech detection models then evolved to use deeper, more complex neural architectures that allow for more comprehensive semantic and syntactic representation. Today, transformer-based pre-trained language models like BERT (Devlin et al., 2019) and its descendants (e.g. Liu et al., 2019c; Conneau et al., 2020; He et al., 2021) are dominating hate speech detection as well as most other NLP tasks. Where logistic regression scored around 74 macro F1 on the widely-used Talat and Hovy (2016) hate speech dataset, Mozafari et al. (2020) score a macro F1 of 88 by simply fine-tuning BERT, which constitutes a vast improvement.

In my thesis, I show that these impressive headline results only tell part of the story. Hate speech detection models have indeed improved substantially over the last years. However, I argue that much of this progress has rested on simplifying assumptions, which, while useful in some settings, we now need to move past in order to develop more truly effective detection models. In particular, I challenge four such assumptions related to hate speech and its detection:

Four Simplifying Assumptions in Hate Speech Research

Assumption 1: Model Accuracy = Model Quality

Reality: Aggregated performance metrics can be useful high-level indicators, but also obscure critical model weaknesses and biases. We need more fine-grained evaluations to more comprehensively characterise model quality.

Assumption 2: Hate Speech Today = Hate Speech Tomorrow

Reality: Hate speech, like all language use, changes over time. Models trained on static data can be effective in many applications, but we also need models that can adapt to language change rather than growing outdated.

Assumption 3: Hate Speech for Me = Hate Speech for You

Reality: Hate speech detection, like many other NLP tasks, is highly subjective. To capture disagreements or apply specific policies, we need frameworks for actively managing subjectivity rather than ignoring it.

ASSUMPTION 4: HATE SPEECH IN ENGLISH = ALL HATE SPEECH

REALITY: Working on a shared language enables progress and collaboration. But hate speech is a global phenomenon, and we need detection models in hundreds more languages to better protect billions of non-English speakers.

Progress in hate speech detection, like in NLP more generally, is guided by how we evaluate models, as common standards of quality incentivise researchers to seek measurable improvements. If we continue to rely on the four simplifying assumptions above, we risk reinforcing standards of quality that are overly aggregated, static, monolithic, and English language-centric. We will not build substantially more robust, more generalisable, or more multilingual hate speech detection models by chasing accuracy gains on the same widely-used English datasets. If, however, we challenge these assumptions and leave them behind – by breaking down model performance, by adapting to language change, by accounting for subjectivity, and by expanding our focus to under-resourced languages – we create new paths for more productive research. Therefore, my thesis has two main goals: 1) to work towards better, more comprehensive standards of quality for hate speech detection models, and 2) to improve models along these standards. Since many of the problems I focus on, such as language change and annotator subjectivity, generalise to broader language modeling, I hope that my thesis can be useful and interesting to hate speech researchers as well as a more general NLP audience.

1.1 THESIS STRUCTURE

The main body of my thesis is organised into five Chapters.

In Chapter 2, I provide background by means of a literature review, focusing on hate speech and modern methods for its automated detection. I discuss different approaches to defining hate speech and introduce the terminology used throughout my thesis. I give a brief history of methods for detecting hate speech from academia, and discuss how they relate to content moderation in industry. Lastly, I summarise the growing literature on evaluating hate speech detection models, and situate my own work within it.

In Chapter 3, I challenge ASSUMPTION 1: MODEL ACCURACY = MODEL QUALITY. Typically, hate speech detection models have been evaluated by measuring their performance on held-out test data using metrics such as accuracy and F1 score. However, this approach makes it difficult to identify specific model weak points. It also risks overestimating generalisable model performance due to increasingly well-evidenced systematic gaps and biases in hate speech datasets. To enable more targeted diagnostic insights, my co-authors and I introduce HateCheck, a suite of functional tests for hate speech detection models. We specify 29 such functional tests, motivated by a review of previous research and a series of interviews with civil society stakeholders. We craft test cases for each test and validate their quality through a structured annotation process. To illustrate HateCheck's utility, we test current academic models as well as two popular commercial models, revealing critical model weaknesses. An edited version of this Chapter, titled "HateCheck - Functional Tests for Hate Speech Detection Models" (Röttger et al., 2021), was published at ACL 2021 (Main Conference).

In Chapter 4, I challenge ASSUMPTION 2: HATE SPEECH TODAY = HATE SPEECH TOMORROW. Language use differs between domains and even within a domain, language use changes over time. For pre-trained language models like BERT, domain adaptation through continued pre-training has been shown to improve performance on in-domain downstream tasks. In this Chapter, my co-author and I investigate whether *temporal adaptation* can bring additional benefits. For this purpose, we first explore a hate speech detection task using temporal data sampled from a far-right social media site. To conduct experiments at a larger scale, we then introduce another corpus of social media comments sampled over three years. This corpus contains unlabelled data for adaptation and evaluation on an upstream masked language modelling task as well as labelled data for fine-tuning and evaluation on a downstream document classification task. We find that temporality matters for both tasks: temporal adaptation improves upstream and temporal fine-tuning downstream task performance. Time-specific models generally perform better on past than on future test sets, which matches evidence on the bursty usage of topical words. However, adapting BERT to time and domain does not improve performance on the downstream task over only adapting to domain. Token-level analysis shows that temporal adaptation captures event-driven changes in language use in the downstream task, but not those

changes that are actually relevant to task performance. Based on our findings, we discuss when temporal adaptation may be more effective. An edited version of this Chapter, titled "Temporal Adaptation of BERT and Performance on Downstream Document Classification: Insights from Social Media" (Röttger and Pierrehumbert, 2021), was published at EMNLP 2021 (Findings).

In Chapter 5, I challenge ASSUMPTION 3: HATE SPEECH FOR ME = HATE SPEECH FOR YOU. Labelled data is the foundation of most NLP tasks. However, labelling data is difficult and there often are diverse valid beliefs about what the correct labels should be. So far, dataset creators have acknowledged annotator subjectivity, but rarely actively managed it in the annotation process. This has led to partly-subjective datasets that fail to serve a clear downstream use. To address this issue, my co-authors and I propose two contrasting paradigms for data annotation. The *descriptive* paradigm encourages annotator subjectivity, whereas the *prescriptive* paradigm discourages it. Descriptive annotation allows for the surveying and modelling of different beliefs, whereas prescriptive annotation enables the training of models that consistently apply one belief. We discuss benefits and challenges in implementing both paradigms, and argue that dataset creators should explicitly aim for one or the other to facilitate the intended use of their dataset. We also conduct an annotation experiment using hate speech data that illustrates the effects of subjectivity as well as the contrast between the two paradigms. An edited version of this Chapter, titled "Two Contrasting Data Annotation Paradigms for Subjective NLP Tasks" (Röttger et al., 2022c), was published at NAACL 2022 (Main Conference).

In Chapter 6, I challenge ASSUMPTION 4: HATE SPEECH IN ENGLISH = ALL HATE SPEECH. Hate speech is a global phenomenon, but most hate speech datasets so far focus on English-language content. This hinders the development of more effective hate speech detection models in hundreds of languages spoken by billions across the world. More data is needed, but annotating hateful content is expensive, time-consuming and potentially harmful to annotators. To mitigate these issues, my co-authors and I explore data-efficient strategies for expanding hate speech detection into under-resourced languages. In a series of experiments with mono- and multilingual models across five non-English languages, we find that 1) a small amount of target-language fine-tuning data is needed to achieve strong performance, 2) the benefits of using more such data decrease exponentially, and 3) initial fine-tuning on readily-available

English data can partially substitute target-language data and improve model generalisability. Based on these findings, we formulate actionable recommendations for hate speech detection in low-resource language settings. An edited version of this Chapter, titled "Data-Efficient Strategies for Expanding Hate Speech Detection into Under-Resourced Languages" (Röttger et al., 2022a), was published at EMNLP 2022 (Main Conference).

Finally, I conclude in Chapter 7 by summarising my work and its impact, discussing outstanding research challenges, and providing an outlook for future research.

NOTE: CHAPTERS VS. ARTICLES Chapters 3, 4, 5 and 6 were each published as separate research articles at NLP conferences. For the purposes of my thesis, I made edits to all four articles. In particular, I edited their Introduction and Conclusion sections to match the overall framing of my thesis. Where appropriate, I updated the Related Works and Discussion sections to include research published after the original articles. The Methodology and Results sections are largely the same as in the original articles, with the exception of Chapter 4, on temporal adaptation, where I included additional experiments on a hate speech detection task.

NOTE: CO-AUTHORSHIP AND FUNDING I was first author on all four research articles included in my thesis. As stipulated by University of Oxford guidelines, my work contributed the majority to each article. For a more detailed breakdown of co-authorship and contributions, see Appendix E. For a funding statement, see Appendix F.

2 Literature Review

2.1 What Is Hate Speech?

There is no unifying definition of hate speech. Within hate speech research, definitions vary (Vidgen et al., 2019; Vidgen and Derczynski, 2020; Poletto et al., 2021), but can broadly be grouped into two categories.

First, there are (quasi-)legal definitions, which define hate speech as a subset of more general abuse, where hate speech is abuse that is targeted at *protected groups* or their members for being a part of that group (e.g. Warner and Hirschberg, 2012; Davidson et al., 2017; Banko et al., 2020). Protected groups, in turn, are commonly defined based on sociodemographic characteristics, like gender or race, by legal standards such as the UK's 2010 Equality Act, the US 1964 Civil Rights Act or the EU's Charter of Fundamental Rights. Conceptually, this aligns hate speech with the legal notion of hate crime. Not all crimes are hate crimes. Similarly, abuse can be deeply hurtful and personally offensive without being hate speech, if it is not related to protected group membership. This is the most common approach for defining hate speech. Content policies for hate speech used by major social media platforms generally follow this pattern, as do most academic definitions. Differences in (quasi-)legal definitions arise because legal standards themselves vary, for example in terms of which groups are considered protected. The US Uniformed Services Employment and Reemployment Rights Act, for instance, protects military veterans, whereas the United Arab Emirates do not consider sexual orientation a protected characteristic (UAE, 2022). Further, illegality is generally seen as too-high a bar for defining hate speech (Vidgen et al., 2019). Most academic definitions of hate speech therefore also include more subtle or implicit content that can be described as *lawful but awful* – the line for which is less clearly drawn and thus often decided on by individual researchers or organisations.

Second, there are more subjective definitions, which define hate speech as content that expresses hate or appears to be motivated by it (e.g. Pitsilis et al., 2018; Kumar et al., 2021; Sap et al., 2022). This necessarily entails more personal interpretation on the part of the hate reader, as

well as speculation about the intent and emotive state of the hate speaker. The same statement may be perceived as hateful by some people and as not hateful by others, depending on extralinguistic factors such as their sociodemographic characteristics (Sap et al., 2019; Salminen et al., 2019; Davani et al., 2021), as I will also show in Chapter 5. This problematises the common practice of labelling hate speech data with majority voting between potentially conflicting subjective perspectives (Plank, 2022). Further, without being anchored to the concept of protected groups, the line between hate speech and non-hateful abuse in the legal sense is blurred. An extreme personal insult, or even just "I hate you", may be considered hate speech in a subjective sense, even if it is not connected in any way to membership in a protected group.

RESEARCH CONTRIBUTIONS My own work engages with these definitional challenges most directly in Chapter 5, where I introduce two contrasting data annotation paradigms for subjective NLP tasks like hate speech detection. My proposed distinction between *prescriptive* annotation, which discourages subjectivity, and *descriptive* annotation, which encourages subjectivity, maps onto the two categories of hate speech definitions outlined above, and provides a framework for operationalising them. In my other thesis work – in Chapters 3, 4 and 6 – I opt for a prescriptive approach grounded in (quasi-)legal definitions of hate speech, which define hate speech as a subset of abuse. The specific definitions I use are stated in each Chapter.

SIDE NOTE: RELATED CONCEPTS Hate speech is closely related to other concepts that seek to describe content which is, in some way and to some people, disruptive or undesirable. Like hate speech, these concepts also lack unifying definitions. *Toxicity* and *offensive language*, for example, are generally used as umbrella terms for various kinds of disruptive content, which include abuse as well as spam, sexually explicit language or the use of profanity. (Zampieri et al., 2019a; Poletto et al., 2021). Banko et al. (2020) introduce *harmful content* as an alternative umbrella term, which includes suicide and self-harm content, but excludes non-abusive profanity. For the purposes of my thesis, when using a prescriptive ap-

proach to defining hate speech, I follow Poletto et al. (2021) in defining hate speech as a target-specific subset of abuse, which is in turn a subset of toxic content.

2.2 How Can We Detect Hate Speech?

Modern research into the automated detection of online hate speech emerged in the early 2010s and has since grown into a large and diverse field of literature. The methods used to detect hate speech, particularly the model architectures, have evolved along with more general trends and advances in language modelling. This evolution can roughly be divided into three time periods.[1]

1) In the early period of modern hate speech detection research, up to circa 2017, the dominant approach was to use simple machine learning models paired with surface-level linguistic features. The most common models were logistic classifiers (Djuric et al., 2015; Talat and Hovy, 2016; Davidson et al., 2017), Naive Bayes (Kwok and Wang, 2013; Liu and Forss, 2014) and tree-based methods (Burnap and Williams, 2014, 2015) as well as support vector machines (Warner and Hirschberg, 2012; Tulkens et al., 2016). Common input features were uni- and bi-grams (e.g. Kwok and Wang, 2013; Liu and Forss, 2014; Burnap and Williams, 2015; Nobata et al., 2016; Davidson et al., 2017) as well as syntactic features such as part-of-speech tags and typed dependencies (e.g. Burnap and Williams, 2014; Gitari et al., 2015; Nobata et al., 2016). Some work also used simple neural word embeddings, then based on word2vec (Mikolov et al., 2013), rather than count-based word representations for input text (e.g. Djuric et al., 2015; Tulkens et al., 2016; Nobata et al., 2016).

These early detection models can provide effective baselines, but suffer from several key weaknesses. They can only account for linguistic features that are explicitly encoded in their inputs (e.g. specific n-grams). They also cannot capture the contextual meaning of specific words (e.g. slurs vs. reclaimed slurs) as well as long-range dependencies across words (e.g. attack and target). As Davidson et al. (2017) argue, such detection models therefore tend to have high false

[1] My thesis, and this literature review, are primarily concerned with text-based hate speech. Detecting hate in other modalities, such as images or videos, as well as multimodal hate, is a smaller but also growing field of research (see e.g. Yang et al., 2019; Kiela et al., 2020; Suryawanshi and Chakravarthi, 2021).

positive rates, since the presence of specific keywords correlated to hate speech (e.g. profanity), can lead to the systematic misclassification of content as hateful even when it is not.

2) The second period of modern hate speech detection research, between 2017 and 2019, followed a general trend in NLP research by introducing deep learning models for content classification. Compared to earlier detection models, these first deep learning models relied on larger, more complex neural architectures, which allow for more comprehensive semantic and syntactic representation, even in longer sequences. However, simple word2vec-like embeddings remained the most common input features. Badjatiya et al. (2017) were among the first to apply convolutional and recurrent neural network architectures (CNNs and RNNs) to the task of hate speech detection, showing that they outperformed simpler classifiers trained on surface-level linguistic features such as n-grams. Subsequent literature expanded on the use of CNNs and RNNs. Gambäck and Sikdar (2017) and Park and Fung (2017), for instance, use CNN variants to outperform logistic classifiers and other simple baselines on data compiled by Talat and Hovy (2016). Pitsilis et al. (2018) use an ensemble of RNNs. Del Vigna et al. (2017) use a long-short term memory network (LSTM), which is a type of RNN, obtaining similar results. Zhang et al. (2018) combine both approaches in a convolutional gated recurrent unit.

Such deep learning models still have clear limitations in their application for hate speech detection. The amount of contextual information they capture is limited, particularly for CNNs with narrow context windows. This complicates context-sensitive distinctions for edge cases between hateful and non-hateful content. Accounting for long-range dependencies remains an issue, even for LSTMs designed to retain information over longer context windows. Furthermore, polysemy, which is particularly relevant to hate speech detection due to the use of coded language (Magu and Luo, 2018) and reclaimed slurs (Sap et al., 2019), poses a challenge when non-contextual embeddings are used. A specific slur, for example, is represented by word2vec as a single embedding, regardless of the linguistic context it appears in.

3) The third and current period of modern hate speech detection started in 2019, with the adoption of pre-trained transformer-based language models like BERT (Devlin et al., 2019). BERT and its descendants improve upon previous deep learning architectures by better accounting for long-range dependencies, contextual information and polysemy. This is achieved through

a bidirectional attention mechanism for computing *contextualised* embeddings, that separately represent the meaning of each (sub-)word in its particular linguistic context (Vaswani et al., 2017). Pre-training is largely unsupervised, using artificial predictions tasks like masked language modelling, and run over large amounts of web data, with the aim of initialising general language capabilities. The pre-trained models are then fine-tuned for particular tasks, like hate speech detection, using labelled data.

BERT-based hate speech detection models immediately achieved state-of-the-art results on several benchmark datasets (Sohn and Lee, 2019; Mozafari et al., 2019; d'Sa et al., 2020) as well as some popular hate speech classification challenges (Zampieri et al., 2019b; Mandl et al., 2019). On a few other hate speech datasets, older deep learning architectures like CNNs and LSTMs paired with additional feature engineering still outperformed vanilla BERT models (Basile et al., 2019). However, the release of even more effective BERT-based architectures, such as RoBERTa (Liu et al., 2019c) and DeBERTa (He et al., 2020), as well as BERT models specifically adapted for hate speech detection (Caselli et al., 2021), cemented the paradigm shift. Today, virtually all state-of-the-art models for hate speech detection and other toxic content classification tasks use pre-trained transformer architectures (see e.g. Pavlopoulos et al., 2021; Vidgen et al., 2021c; Nozza et al., 2022; Lees et al., 2022). These architectures will likely continue to improve for some time, because of benefits to scale in pre-training data and model parameters (Kaplan et al., 2020; Hoffmann et al., 2022; OpenAI, 2023).

RESEARCH CONTRIBUTIONS Rather than introducing novel architectures for hate speech detection or more general language modelling, my thesis work seeks to improve the performance of existing architectures in under-researched settings through better pre-training and fine-tuning approaches. Specifically, in Chapter 4, I evaluate the benefits of continued pre-training on time-segmented data for adapting models to language change over time. In Chapter 6, I explore data-efficient methods for fine-tuning transformer models to improve performance in low-resource languages.

SIDE NOTE: CONTENT MODERATION SYSTEMS Hate speech detection models, as they are discussed here, play an important role in content moderation on large social media platforms like Facebook and Twitter, but they are only one part of larger content moderation systems, which also utilise human review, keyword filters and hash lists (Gillespie, 2018). Other platforms, like Mastodon or Telegram, appear to use very little automated detection, aiming for community-based moderation or a near-total absence of moderation instead. My thesis is not about whether and when we should use hate speech detection models, or algorithmic content moderation more generally (see Gorwa et al., 2020; Gillespie, 2020, 2022). Rather, my thesis rests on the assumption, backed by fact, that flawed hate speech detection models are used to moderate content at scale right now, and that therefore we need to 1) understand the strengths and weaknesses of these models, and 2) improve them where possible, so as to make content moderation overall more equitable, efficient and effective. This stands to benefit platform users, who will be better protected, as well as platform moderators, who will be less exposed to potentially harmful content. One limitation of my work, and a challenge for hate speech research as a whole, is the potential disconnect between academic research into hate speech detection models, which I focus on, and highly intransparent industry practices around content moderation systems. I discuss this in more detail in Chapter 7.

2.3 HOW DO WE TEST HATE SPEECH DETECTION MODELS?

Hate speech detection has been adopted by the NLP community as a classification task concerned with identifying examples of hate speech in a larger corpus of documents. This aligns hate speech detection with other, established classification tasks like sentiment or emotion detection as well as spam detection. Accordingly, most hate speech research so far has used the same, established metrics to evaluate model performance, and by extension model quality. Most

commonly, performance is measured on a single dataset using a single metric, which can then be compared across models. Model accuracy, for example, calculates the percentage of correctly-classified entries. F1 scores for each class calculate the harmonic mean of model precision and recall. The most accurate or highest-scoring model is then said to be the state of the art.

By any of these metrics, the methodological advancements described in Section 2.2 above led to drastic improvements in model quality. Mozafari et al. (2020), for example, used a simple BERT-based method to achieve a macro F1 score of 88 on the widely-used Davidson et al. (2017) dataset, which leaves limited room for further improvement. However, the real-world performance of systems powered by hate speech detection models makes it clear that hate speech detection is far from perfect. Automated content moderation is still prone to clear errors, which journalists and activists are regularly reporting on. Anti-racist Facebook posts (Guynn, 2019) and innocuous football tweets (Langran, 2021), for example, are mistakenly taken down, while misogynistic and homophobic content can remain on Facebook for weeks (Scott, 2021). This mismatch in academic results and real-world performance needs explanation, to support the development of more practically effective hate speech detection models. In recent years, hate speech research has therefore begun to focus more on model evaluation.

A first stream of research is concerned with the generalisability of hate speech detection models. Nejadgholi and Kiritchenko (2020), Samory et al. (2021) and Yin and Zubiaga (2021) all use cross-dataset evaluations to show that detection models trained on one dataset generalise poorly to other datasets, because of dataset idiosyncrasies picked up during training. Relatedly, Park et al. (2018), Dixon et al. (2018), Kennedy et al. (2020) and Chuang et al. (2021) show that models are overly sensitive to group identifiers like "black" or "gay", and tend to classify their use as hateful regardless of linguistic context. More recently, these findings have motivated research into techniques aimed at mitigating these unintended biases, for example through data augmentation (Bartl et al., 2020; Sharma et al., 2020; de Vassimon Manela et al., 2021), as well as entropy-based regularisation of the explanation-based importance of group identifiers (Kennedy et al., 2020), attention (Attanasio et al., 2022) and loss (Tiwari et al., 2022).

A second stream of research is more directly concerned with evaluating model performance as well as specific model strengths and weaknesses using individual datasets. My own work on

HateCheck (Chapter 3) introduced functional testing – a concept from software engineering that was first applied to NLP models by Ribeiro et al. (2020) – for hate speech detection, by using hand-crafted test cases to diagnose model behaviour on different kinds of hate and non-hate.[2] Kirk et al. (2022c) applied the same framework for emoji-based hate. Manerba and Tonelli (2021) provide smaller-scale functional tests for abuse detection systems. Other research has instead collected real-world examples of hate and annotated them for more fine-grained labels, such as the hate target, to enable more comprehensive error analysis (e.g. Mathew et al., 2021; Vidgen et al., 2021c; Mollas et al., 2022). Instead of creating a static dataset, Calabrese et al. (2021) devise a hate speech-specific data augmentation technique based on simple heuristics to create additional test cases for a given set of training data.

RESEARCH CONTRIBUTIONS Creating better, more comprehensive standards of model quality is one of the main goals of my thesis. Most relevantly, Chapter 3 introduces HateCheck as a new, fine-grained evaluation tool for hate speech detection models. The other Chapters each diagnose specific issues with current hate speech datasets and detection models that relate to the simplifying assumptions they are built on. Chapter 4 highlights how the static nature of hate speech datasets is at odds with language change, based on poor cross-temporal performance of classification models. Chapter 5 evidences the subjectivity of hate speech through an annotation experiment, therefore problematising the use of monolithic hate speech datasets with single majority labels. Chapter 6 shows the importance of non-English resources for multilingual hate speech detection and thus challenges current English language-centric development and evaluation practices.

[2]In Röttger et al. (2022b), we expanded the original English HateCheck to ten additional languages. This work was not part of my DPhil and is therefore not included in my thesis.

3 HATECHECK - FUNCTIONAL TESTS FOR HATE SPEECH DETECTION MODELS

Authors: **Paul Röttger** (University of Oxford, Alan Turing Institute), **Bertie Vidgen** (Alan Turing Institute), **Dong Nguyen** (Utrecht University), **Zeerak Talat** (University of Sheffield), **Helen Margetts** (University of Oxford, Alan Turing Institute), and **Janet B. Pierrehumbert** (University of Oxford)[3]

Edited Chapter published at **ACL 2021** (Main Conference) as Röttger et al. (2021).

3.1 INTRODUCTION

In modern hate speech research, hate speech detection models have primarily been evaluated by measuring held-out performance on a small set of widely-used hate speech datasets (particularly Talat and Hovy, 2016; Davidson et al., 2017; Founta et al., 2018). This is what I denote as ASSUMPTION 1: MODEL ACCURACY = MODEL QUALITY. However, recent work has highlighted the limitations of this evaluation paradigm. Aggregate performance metrics offer limited insight into specific model weaknesses (Wu et al., 2019). Further, if there are systematic gaps and biases in training data, models may perform deceptively well on corresponding held-out test sets by learning simple decision rules rather than encoding a more generalisable understanding of the task (e.g. Niven and Kao, 2019; Geva et al., 2019; Shah et al., 2020). The latter issue is particularly relevant to hate speech detection since current hate speech datasets vary in data source, sampling strategy and annotation process (Vidgen and Derczynski, 2020; Poletto et al., 2021), and are known to exhibit annotator biases (Talat, 2016; Talat et al., 2018; Sap et al., 2019; Kumar et al., 2021; Sap et al., 2022) as well as topic and author biases (Wiegand et al., 2019; Nejadgholi and Kiritchenko, 2020). Correspondingly, models trained on such datasets have been shown to be overly sensitive to lexical features such as group identifiers (Park et al., 2018; Dixon et al., 2018; Kennedy et al., 2020; Attanasio et al., 2022), and to generalise poorly

[3]All author affiliations listed throughout my thesis are those at the time of the original article's publication.

to other datasets (Nejadgholi and Kiritchenko, 2020; Samory et al., 2021; Yin and Zubiaga, 2021). Therefore, held-out performance on current hate speech datasets is an incomplete and potentially misleading measure of model quality.

To enable more targeted diagnostic insights, this Chapter introduces HateCheck, a suite of functional tests for hate speech detection models. Functional testing is a testing framework from software engineering that assesses different functionalities of a given model by comparing actual and desired model outputs on sets of targeted test cases, similarly to unit testing (Beizer, 1995). Ribeiro et al. (2020) show how such a framework can be used for structured model evaluation across many different NLP tasks.

HateCheck covers 29 model functionalities, the selection of which we motivate through a series of interviews with civil society stakeholders and a review of hate speech research. Each functionality is tested by a separate functional test. We create 18 functional tests corresponding to distinct expressions of hate. The other 11 functional tests are non-hateful contrasts to the hateful cases. For example, we test non-hateful reclaimed uses of slurs as a contrast to their hateful use. Such tests are particularly challenging to models relying on overly simplistic decision rules and thus enable more accurate evaluation of true model functionalities (Gardner et al., 2020). For each functional test, we hand-craft sets of targeted test cases with clear gold standard labels, which we validate through a prescriptive annotation process.[4]

HateCheck is broadly applicable across English-language hate speech detection models. We demonstrate its utility as a diagnostic tool by evaluating two BERT models (Devlin et al., 2019), which have achieved near state-of-the-art performance on hate speech datasets (Tran et al., 2020), as well as two commercial models – Google Jigsaw's Perspective and Two Hat's SiftNinja. When tested with HateCheck, all models appear overly sensitive to specific keywords such as slurs. They consistently misclassify negated hate, counter speech and other non-hateful contrasts to hateful phrases. Further, the BERT models are biased in their performance across target groups, misclassifying more content directed at some groups (e.g. women) than at others. For practical applications such as content moderation and further research use, these are critical model weaknesses. In line with the overall goals of my thesis, we hope that by revealing

[4]All HateCheck test cases and annotations are available on https://github.com/paul-rottger/hatecheck-data.

such weaknesses, HateCheck can help researchers address them, and thus play a key role in the development of better hate speech detection models.

DEFINITION OF HATE SPEECH We draw on previous definitions of hate speech (Warner and Hirschberg, 2012; Davidson et al., 2017) as well as comprehensive typologies of abusive content (Vidgen et al., 2019; Banko et al., 2020) to define hate speech as *abuse that is targeted at a protected group or at its members for being a part of that group*. We define protected groups based on age, disability, gender identity, familial status, pregnancy, race, national or ethnic origins, religion, sex or sexual orientation, which broadly reflects international legal consensus (particularly the UK's 2010 Equality Act, the US 1964 Civil Rights Act and the EU's Charter of Fundamental Rights). Based on these (quasi-)legal definitions, we approach hate speech detection as the binary classification of content as either hateful or non-hateful. Other work has further differentiated between different types of hate and non-hate (e.g. Founta et al., 2018; Salminen et al., 2018; Zampieri et al., 2019a), but such taxonomies can be collapsed into a binary distinction and are thus compatible with HateCheck.

3.2 CONSTRUCTING HATECHECK

3.2.1 DEFINING MODEL FUNCTIONALITIES

In software engineering, a program has a certain *functionality* if it meets a specified input/output behaviour, where actual output matches desired output (ISO, 2017). Accordingly, we operationalise a functionality of a hate speech detection model as its ability to provide a specified classification (hateful or non-hateful) for test cases in a corresponding functional test. For instance, a model might correctly classify hate expressed using profanity (e.g "F*ck all Black people") but misclassify non-hateful uses of profanity (e.g. "F*cking hell, what a day"), which is why we test them as separate functionalities. Since both functionalities relate to profanity usage, we group them into a common *functionality class*.

3.2.2 Selecting Functionalities for Testing

To generate an initial list of 59 functionalities, we reviewed previous hate speech detection research and interviewed civil society stakeholders.

Review of Previous Research We identified different types of hate in taxonomies of abusive content (e.g. Zampieri et al., 2019a; Banko et al., 2020; Kurrek et al., 2020). We also identified likely model weaknesses based on error analyses (e.g. Davidson et al., 2017; van Aken et al., 2018; Vidgen et al., 2020) as well as review articles and commentaries (e.g. Schmidt and Wiegand, 2017; Fortuna and Nunes, 2018; Vidgen et al., 2019). For example, hate speech detection models have been shown to struggle with correctly classifying negated phrases such as "I don't hate trans people" (Hosseini et al., 2017; Dinan et al., 2019). We therefore included functionalities for negation in hateful and non-hateful content.

Interviews We interviewed 21 employees from 16 British, German and American NGOs whose work directly relates to online hate. Most of the NGOs are involved in monitoring and reporting online hate, often with "trusted flagger" status on platforms such as Twitter and Facebook. Several NGOs provide legal advocacy and victim support or otherwise represent communities that are often targeted by online hate, such as Muslims or LGBT+ people. The vast majority of interviewees do not have a technical background, but extensive practical experience engaging with online hate and content moderation systems. They have a variety of ethnic and cultural backgrounds, and most of them have been targeted by online hate themselves.

The interviews were semi-structured. In a typical interview, we would first ask open-ended questions about online hate (e.g. "What do you think are the biggest challenges in tackling online hate?") and then about hate speech detection models, particularly their perceived weaknesses (e.g. "What sort of content have you seen moderation systems get wrong?") and potential improvements, unbounded by technical feasibility (e.g. "If you could design an ideal hate detection system, what would it be able to do?"). Using a grounded theory approach (Corbin and Strauss, 1990), we identified emergent themes in the interview responses and translated them

into model functionalities. For example, several interviewees raised concerns around the misclassification of counter speech, i.e. direct responses to hateful content (e.g. I4: "people will be quoting someone, calling that person out [...] but that will get picked up by the system").[5] We therefore included functionalities for counter speech that quotes or references hate.

SELECTION CRITERIA From the initial list of 59 functionalities, we select those in HateCheck based on two practical considerations.

First, we restrict HateCheck's scope to individual English language text documents. This is due to practical constraints, and because most hate speech detection models are developed for such data (Vidgen and Derczynski, 2020; Poletto et al., 2021). Thus, HateCheck does not test functionalities that relate to other modalities (e.g. images) or languages, or that require context (e.g. conversational or social) beyond individual documents.

Second, we only test functionalities for which we can construct test cases with clear gold standard labels. Therefore, we do not test functionalities that lack broad consensus in our interviews and the literature regarding what is and is not hateful. The use of humour, for instance, has been highlighted as an important challenge for hate speech research (van Aken et al., 2018; Qian et al., 2018; Vidgen et al., 2020). However, whether humorous statements qualify as hate speech is heavily contingent on normative claims (e.g. I5: "it's a value judgment thing"), which is why we do not test them in HateCheck.

3.2.3 FUNCTIONAL TESTS IN HATECHECK

HateCheck comprises 29 functional tests grouped into 11 classes. Each test evaluates one functionality and is associated with one gold standard label (hateful or non-hateful). Each functional test has a set of corresponding test cases.

18 functional tests for hateful content in HateCheck cover **distinct expressions of hate**. They are distinct in the sense that we minimise overlap between them, for instance by testing slurs

[5] When quoting anonymised responses throughout this Chapter, we identify each interview participant by a unique ID. We cannot release full interview transcripts due to the sensitive nature of work in this area, the confidentiality terms agreed with our participants and our ethics clearance.

("f*g") and profanity ("f*ck") in separate functional tests rather than jointly ("f*cking f*g"), so that each test isolates one particular type of expression.

The other 11 functional tests for non-hateful content cover **contrastive non-hate**, i.e. content which shares linguistic features with hateful expressions. The challenges posed by such content are a key theme in our interviews and the literature. We construct every non-hateful test case as a direct contrast to a hateful test case, making only minimal changes. For instance, "I love immigrants" is a test case in **F19**: positive statements using a protected group identifier. It directly contrasts the test case "I hate immigrants" in **F1**: strong negative emotions explicitly expressed about a protected group.

In the following, we give a brief overview of the different functional tests in HateCheck. Table 1 provides corresponding example test cases. Each individual test is grounded in direct references to previous work and/or our interviews. These references are detailed in Appendix A.3.

DISTINCT EXPRESSIONS OF HATE HateCheck tests different types of derogatory hate speech (**F1-4**) and hate expressed through threatening language (**F5/6**). It tests hate expressed using slurs (**F7**) and profanity (**F10**). It also tests hate expressed through pronoun reference (**F12/13**), negation (**F14)** and phrasing variants, specifically questions and opinions (**F16/17**). Lastly, it tests hate containing spelling variations such as missing characters or leet speak (**F25-29**).

CONTRASTIVE NON-HATE HateCheck tests non-hateful contrasts for slurs, particularly slur homonyms and reclaimed slurs (**F8/9**), as well as for profanity (**F11**). It tests non-hateful contrasts that use negation, i.e. negated hate (**F15**). It also tests non-hateful contrasts around protected group identifiers (**F18/19**). It tests contrasts in which hate speech is quoted or referenced to non-hateful effect, specifically *counter speech*, i.e. direct responses to hate speech which seek to act against it (**F20/21**). Lastly, it tests non-hateful contrasts which target out-of-scope entities such as generic individuals and objects (**F22-24**) rather than a protected group.

	Functionality	Example Test Case	Gold Label	n	Accuracy (%)			
					B-D	B-F	P	SN
Derogation	**F1**: Expression of strong negative emotions (explicit)	"I hate [IDENTITY]." "You [IDENTITY] make me sick."	hateful	140	88.6	90.7	**98.6**	*7.9*
	F2: Description using very negative attributes (explicit)	"[IDENTITY] are disgusting." "All [IDENTITY] like them are vile."	hateful	140	88.6	84.3	**95.7**	*0.0*
	F3: Dehumanisation (explicit)	"[IDENTITY] are scum." "[IDENTITY] are a pest."	hateful	140	91.4	80.7	**98.6**	*10.0*
	F4: Implicit derogation	"[IDENTITY] belong in a zoo." "I'd rather die than date [IDENTITY]."	hateful	140	71.4	61.4	**85.0**	*0.0*
Threatening language	**F5**: Direct threat	"I am going to shoot the next [IDENTITY] I see."	hateful	133	87.2	75.9	**100.0**	*4.5*
	F6: Threat as normative statement	"We should get rid of [IDENTITY] once and for all."	hateful	140	91.4	83.6	**100.0**	*2.1*
Slur usage	**F7**: Hate expressed using slur	"You are just a [SLUR] to me."	hateful	144	60.4	*41.0*	**66.0**	50.0
	F8: Non-hateful homonyms of slurs	"I had this queer feeling we were being watched."	non-hate	30	66.7	**70.0**	63.3	*33.3*
	F9: Reclaimed slurs	"I'm the freshest n*gga in the room."	non-hate	81	***39.5***	*33.3*	*28.4*	*18.5*
Profanity usage	**F10**: Hate expressed using profanity	"These [IDENTITY] should shut the f*ck up once and for all."	hateful	140	82.9	72.9	**100.0**	*1.4*
	F11: Non-hateful use of profanity	"I wish someone could get me the f*ck out of here."	non-hate	100	99.0	**100.0**	98.0	**100.0**
Pronoun reference	**F12**: Hate expressed through reference in subsequent clauses	"I have met many [IDENTITY] and I hate every single one of them."	hateful	140	87.1	80.7	**99.3**	*0.7*
	F13: Hate expressed through reference in subsequent sentences	"Of course they are stupid. They are just [IDENTITY] after all."	hateful	133	85.7	70.7	**100.0**	*0.0*
Negation	**F14**: Hate expressed using negated positive statement	"[IDENTITY] lives do not matter."	hateful	140	85.0	60.7	**96.4**	*2.9*
	F15: Non-hate expressed using negated hateful statement	"No [IDENTITY] deserves to die." "We shouldn't let [IDENTITY] suffer."	non-hate	133	*12.8*	*12.0*	*3.8*	**96.2**
Phrasing	**F16**: Hate phrased as a question	"Why is it that all [IDENTITY] are repulsive?"	hateful	140	80.7	75.0	**99.3**	*9.3*
	F17: Hate phrased as an opinion	"I think that [IDENTITY] should all be killed."	hateful	133	85.7	75.9	**98.5**	*2.3*
Non-hate grp. ident.	**F18**: Neutral statements using protected group identifiers	"We are a group of [IDENTITY]." "I live with two [IDENTITY]."	non-hate	126	*20.6*	58.7	84.1	**100.0**
	F19: Positive statements using protected group identifiers	"I love [IDENTITY]." "[IDENTITY] are great."	non-hate	189	*21.7*	52.9	54.0	**100.0**
Counter speech	**F20**: Denouncements of hate that quote it	"Statements like '[IDENTITY] are scum' are deeply hurtful."	non-hate	173	*26.6*	*32.9*	*15.6*	**79.8**
	F21: Denouncements of hate that make direct reference to it	"You have to stop calling [IDENTITY] disgusting."	non-hate	141	*29.1*	*29.8*	*18.4*	**79.4**
Abuse against non-prot. targets	**F22**: Abuse targeted at objects	"I really can't stand cauliflower." "All cocktails like these are vile"	non-hate	65	87.7	84.6	95.4	**100.0**
	F23: Abuse targeted at individuals (not as member of a prot. group)	"You make me sick." "I will hurt you."	non-hate	65	*27.7*	55.4	84.6	**100.0**
	F24: Abuse targeted at non-protected groups (e.g. professions)	"Artists are parasites to our society." "Death to all capitalists."	non-hate	62	*35.5*	59.7	62.9	**98.4**
Spelling variations	**F25**: Swaps of adjacent characters	"I haet [IDENTITY]."	hateful	133	69.9	58.6	**88.7**	*11.3*
	F26: Missing characters	"[IDENTITY] are a cancr."	hateful	140	59.3	*47.9*	**74.3**	*0.0*
	F27: Missing word boundaries	"Deathto all [IDENTITY]."	hateful	141	68.1	51.1	**80.1**	*13.5*
	F28: Added spaces between chars	"I want to k i l l all [IDENTITY]."	hateful	173	*43.9*	*37.6*	**74.0**	*22.5*
	F29: Leet speak spellings	"[IDENTITY] lives aren't worth sh1t."	hateful	173	*48.0*	*43.9*	**68.2**	*16.2*

Table 1: HateCheck covers 29 functionalities in 11 classes with a total of n = 3,728 test cases. 68.8% of cases (2,563 in 18 functional tests) are labelled hateful, 31.2% (1,165 in 11 functional tests) are labelled non-hateful. The right-most columns report accuracy (%) on each functional test for the models described in §3.3.1. Best performance on each functional test is **bolded**. Below random choice performance (<50%) is highlighted in *cursive red*.

3.2.4 Generating Test Cases

For each functionality in HateCheck, we hand-craft sets of test cases – short English-language text documents that clearly correspond to just one gold standard label. Within each functionality, we aim to use diverse vocabulary and syntax to reduce similarity between test cases, which Zhou et al. (2020) suggest as a likely cause of performance instability for diagnostic datasets.

To generate test cases at scale, we use templates (Dixon et al., 2018; Garg et al., 2019; Ribeiro et al., 2020), in which we replace tokens for protected group identifiers (e.g. "I hate [IDENTITY].") and slurs (e.g. "You are just a [SLUR] to me."). This also ensures that HateCheck has an equal number of cases targeted at different protected groups.

HateCheck covers seven protected groups: women (gender), trans people (gender identity), gay people (sexual orientation), Black people (race), disabled people (disability), Muslims (religion) and immigrants (national origin). For details on which slurs are covered by HateCheck and how they were selected, see Appendix A.4.

In total, we generate 3,901 cases, 3,495 of which come from 460 templates. The other 406 cases do not use template tokens (e.g. "Sh*t, I forgot my keys") and are thus crafted individually. The average length of cases is 8.87 words (std. dev. = 3.33) or 48.26 characters (std. dev. = 16.88). 2,659 of the 3,901 cases (68.2%) are hateful and 1,242 (31.8%) are non-hateful.

Secondary Labels In addition to the primary label (hateful or non-hateful) we provide up to two secondary labels for all cases. For cases targeted at or referencing a particular protected group, we provide a label for the group that is targeted. For hateful cases, we also label whether they are targeted at a group in general or at individuals, which is a common distinction in taxonomies of abuse (e.g. Talat et al., 2017; Zampieri et al., 2019a).

3.2.5 Validating Test Cases

To validate gold standard primary labels of test cases in HateCheck, we recruited and trained ten annotators for prescriptive annotation according to our definition of hate speech.[6] In addition to

[6]For information on annotator training, background and demographics, see the data statement in Appendix A.2.

the binary annotation task, we also gave annotators the option to flag cases as unrealistic (e.g. nonsensical) to further confirm data quality. Each annotator was randomly assigned approximately 2,000 test cases, so that each of the 3,901 cases was annotated by exactly five annotators. We use Fleiss' Kappa to measure inter-annotator agreement (Hallgren, 2012) and obtain a score of 0.93, which indicates "almost perfect" agreement (Landis and Koch, 1977).

For 3,879 (99.4%) of the 3,901 cases, at least four out of five annotators agreed with our gold standard label. For 22 cases, agreement was less than four out of five. To ensure that the label of each HateCheck case is unambiguous, we exclude these 22 cases. We also exclude all cases generated from the same templates as these 22 cases to avoid biases in target coverage, as otherwise hate against some protected groups would be less well represented than hate against others. In total, we exclude 173 cases, reducing the size of the dataset to 3,728 test cases.[7] Only 23 cases were flagged as unrealistic by one annotator, and none were flagged by more than one annotator. Thus, we do not exclude any test cases for being unrealistic.

3.3 Testing Models with HateCheck

3.3.1 Model Setup

As a suite of functional tests, HateCheck is broadly applicable across English-language hate speech detection models. Users can evaluate different architectures trained on different datasets and even commercial models for which public information on architecture and training data is limited. To illustrate HateCheck's applicability, we test four different detection models.

Pre-Trained Transformer Models We test an uncased BERT-base model (Devlin et al., 2019), which has been shown to achieve near state-of-the-art performance on several abuse detection tasks (Tran et al., 2020). We fine-tune this BERT model on two widely-used hate speech datasets from Davidson et al. (2017) and Founta et al. (2018).

The Davidson et al. (2017) dataset contains 24,783 tweets annotated as either *hateful*, *offensive* or *neither*. The Founta et al. (2018) dataset comprises 99,996 tweets annotated as *hateful*,

[7] We make data on annotation outcomes available for all cases we generated, including the ones not in HateCheck.

abusive, *spam* and *normal*. For both datasets, we collapse labels other than hateful into a single non-hateful label to match HateCheck's binary format. This is aligned with the original multi-label setup of the two datasets. Davidson et al. (2017), for instance, explicitly characterise offensive content in their dataset as non-hateful. Respectively, hateful cases make up 5.8% and 5.0% of the datasets. Details on both datasets and text pre-processing steps can be found in Appendix A.5.

In the following, we denote BERT fine-tuned on binary Davidson et al. (2017) data by **B-D** and BERT fine-tuned on binary Founta et al. (2018) data by **B-F**. To account for class imbalance in the training data, we use class weights emphasising the hateful minority class (He and Garcia, 2009). For both datasets, we use a stratified 80/10/10 train/dev/test split. Macro F1 on the held-out test sets is 70.8 for **B-D** and 70.3 for **B-F**.[8] We list details on model training and hyperparameters in Appendix A.6.

COMMERCIAL MODELS We test Google Jigsaw's Perspective (**P**) and Two Hat's SiftNinja (**SN**).[9] Both are popular models for content moderation developed by major tech companies that can be accessed by registered users via an API. For both models, it is unclear what data they were (or were not) trained on, and which model architectures they use.

For a given input text, **P** provides percentage scores across attributes such as "toxicity" and "profanity". We use "identity attack", which aims at identifying "negative or hateful comments targeting someone because of their identity" and thus aligns closely with our definition of hate speech (§3.1). We tested **P** in December 2020.

For **SN**, we use its 'hate speech' attribute ("attacks [on] a person or group on the basis of personal attributes or identities"), which distinguishes between 'mild', 'bad', 'severe' and 'no' hate, mapped onto a numerical scale of zero to one, where 'no' corresponds to zero and all other classes correspond to values larger than 0.5. We tested **SN** in January 2021.

[8] For better comparability to previous work, we also fine-tuned unweighted versions of our models on the original multiclass **D** and **F** data. Their performance matches high-performing systems reported elsewhere (Mozafari et al., 2019; Cao et al., 2020). Details in Appendix A.7.

[9] www.perspectiveapi.com and www.siftninja.com. SiftNinja was shut down in 2022.

A NOTE ON CALIBRATION All four models we test predict percentage scores, i.e. scores between zero and one, where one is the most confident hateful prediction. We convert these percentage scores to binary predictions using a cutoff of 50% for all models. For all our experiments, we assess model performance using accuracy, i.e. the proportion of correctly classified test cases. Trivially, setting a lower cutoff would make the same model more accurate on hateful cases and less accurate on non-hateful cases in HateCheck. Setting a higher cutoff would do the opposite. We choose a 50% cutoff to reflect standard practices in hate speech detection (Vidgen and Derczynski, 2020; Poletto et al., 2021) as well as the default model behaviour in popular programming libraries like HuggingFace's `transformers` (Wolf et al., 2020). Essentially, the 50% cutoff reflects how we would expect the models that we test to be most commonly used, and this is what we want to capture in these experiments, where the primary goal is to illustrate HateCheck's application. Furthermore, calibrating the models to optimise performance on HateCheck would require information about HateCheck, such as its label distribution or even gold standard labels, and thus invalidate HateCheck as a truly external diagnostic tool.

NAIVE BASELINES We compare performance of all models to a random choice baseline, which classifies entries as hateful or non-hateful with equal probability. The expected accuracy of such a baseline is 50%, regardless of the class distribution of the data it is applied to. As with the calibration of the other models we test, the rationale for this random choice baseline is that it requires no ex-ante knowledge of HateCheck, its labels or class distribution. It is also easy to compare to across all our experiments. Trivially, an always-hateful baseline would be perfectly accurate on hateful cases and perfectly inaccurate on non-hateful cases. The opposite would be true for a never-hateful baseline. Therefore, model performance on a given functional test in HateCheck should only ever be considered an indicator of model quality in the context of performance on other functional tests. When reporting accuracy in tables, we **bolden** the best performance, i.e. the highest accuracy across models and highlight accuracy below the random choice baseline, i.e. 50% for our binary task, in *cursive red*.

3.3.2 RESULTS

PERFORMANCE ACROSS LABELS All models show clear performance deficits when tested on hateful and non-hateful cases in HateCheck (Table 2). **B-D**, **B-F** and **P** are relatively more accurate on hateful cases but misclassify most non-hateful cases. In total, **P** performs best, because it is substantially more accurate on hateful cases than **B-F** while being similarly accurate on non-hateful cases. **SN** performs worst and is strongly biased towards classifying all cases as non-hateful, making it highly accurate on non-hateful cases but misclassify most hateful cases.

Label	**n**	**B-D**	**B-F**	**P**	**SN**	rnd	ah	nh
Hateful	2,563	75.5	65.5	**89.5**	*9.0*	50.0	100	*0.0*
Non-hateful	1,165	*36.0*	*48.5*	*48.2*	**86.6**	50.0	*0.0*	100
Total	3,728	63.2	60.2	**76.6**	*33.2*	50.0	68.8	*31.2*

Table 2: Accuracy (%) by test case label across models. The right-hand side of the table shows three naive baselines (**ran**dom choice, **a**lways-**h**ateful and **n**ever-**h**ateful). We use the random choice baseline (accuracy = 50%) for comparison throughout our experiments.

PERFORMANCE ACROSS FUNCTIONAL TESTS Evaluating models on each functional test (Table 1) reveals specific model weaknesses.

B-D and **B-F**, respectively, are less than 50% accurate on 8 and 4 out of the 11 functional tests for non-hate in HateCheck. In particular, the models misclassify most cases of reclaimed slurs (**F9**, 39.5% and 33.3% correct), negated hate (**F15**, 12.8% and 12.0% correct) and counter speech (**F20/21**, 26.6%/29.1% and 32.9%/29.8% correct). **B-D** is slightly more accurate than **B-F** on most functional tests for hate while **B-F** is more accurate on most tests for non-hate. Both models generally do better on hateful than non-hateful cases, although they struggle, for instance, with spelling variations, particularly added spaces between characters (**F28**, 43.9% and 37.6% correct) and leet speak spellings (**F29**, 48.0% and 43.9% correct).

P performs better than **B-D** and **B-F** on most functional tests. It is over 95% accurate on 11 out of 18 functional tests for hate and substantially more accurate than **B-D** and **B-F** on spelling variations (**F25-29**). However, it performs even worse than **B-D** and **B-F** on non-

hateful functional tests for reclaimed slurs (**F9**, 28.4% correct), negated hate (**F15**, 3.8% correct) and counter speech (**F20/21**, 15.6%/18.4% correct).

Due to its tendency to classify all cases as non-hateful, **SN** misclassifies most hateful cases and is near-perfectly accurate on non-hateful functional tests. Exceptions to the latter are counter speech (**F20/21**, 79.8%/79.4% correct) and non-hateful slur usage (**F8/9**, 33.3%/18.5% correct), which are still misclassified at a substantial rate.

PERFORMANCE ON INDIVIDUAL FUNCTIONAL TESTS Individual functional tests can be split out to show more granular model weaknesses. To illustrate, Table 3 reports model accuracy on test cases for non-hateful reclaimed slurs (**F9**) grouped by the reclaimed slur that is used.

Reclaimed Slur	**n**	**B-D**	**B-F**	**P**	**SN**
N*gga	19	**89.5**	*0.0*	*0.0*	*0.0*
F*g	16	*0.0*	***6.2***	*0.0*	*0.0*
F*ggot	16	*0.0*	***6.2***	*0.0*	*0.0*
Q*eer	15	*0.0*	73.3	**80.0**	*0.0*
B*tch	15	**100.0**	93.3	73.3	**100.0**

Table 3: Model accuracy (%) on test cases for reclaimed slurs (**F9**, non-hateful) by slur.

Performance varies across models and is strikingly poor on individual slurs. **B-D** misclassifies all instances of "f*g", "f*ggot" and "q*eer". **B-F** and **P** perform better for "q*eer", but fail on "n*gga". **SN** fails on all cases but reclaimed uses of "b*tch".

PERFORMANCE ACROSS TARGET GROUPS HateCheck can test whether models exhibit 'unintended biases' (Dixon et al., 2018) in target coverage by evaluating their performance on cases which target different groups. To illustrate, Table 4 shows model accuracy on all test cases created from [IDENTITY] templates, which only differ in the group identifier. Larger variance in accuracy across groups corresponds to a stronger bias in target coverage.

B-D misclassifies test cases targeting women twice as often as those targeted at other groups. **B-F** also performs relatively worse for women and fails on most test cases targeting disabled

Targeted Group	n	B-D	B-F	P	SN
Women	421	*34.9*	52.3	**80.5**	*23.0*
Trans people	421	69.1	69.4	**80.8**	*26.4*
Gay people	421	73.9	74.3	**80.8**	*25.9*
Black people	421	69.8	72.2	**80.5**	*26.6*
Disabled people	421	71.0	*37.1*	**79.8**	*23.0*
Muslims	421	72.2	73.6	**79.6**	*27.6*
Immigrants	421	70.5	58.9	**80.5**	*25.9*

Table 4: Model accuracy (%) on cases generated from [IDENTITY] templates, by target group.

people, while being more than 70% accurate on test cases targeting other groups. By contrast, **P** is consistently around 80% accurate and **SN** around 25% accurate across target groups.

3.3.3 Discussion

HateCheck reveals functional weaknesses in all four models that we test.

First, all models are overly sensitive to specific keywords in at least some contexts. **B-D**, **B-F** and **P** perform well for both hateful and non-hateful cases of profanity (**F10/11**), which shows that they can distinguish between different uses of certain profanity terms. However, all models perform very poorly on reclaimed slurs (**F9**) compared to hateful slurs (**F7**). Thus, it appears that the models to some extent encode overly simplistic keyword-based decision rules (e.g. that slurs are hateful) rather than capturing the relevant linguistic phenomena (e.g. that slurs can have non-hateful reclaimed uses).

Second, **B-D**, **B-F** and **P** struggle with non-hateful contrasts to hateful phrases. In particular, they misclassify most cases of negated hate (**F15**) and counter speech (**F20/21**). Thus, they appear to not sufficiently register linguistic signals that reframe hateful phrases into clearly non-hateful ones (e.g. "No Muslim deserves to die").

Third, **B-D** and **B-F** are biased in their target coverage, classifying hate directed against some protected groups (e.g. women) less accurately than equivalent cases directed at others (Table 4).

For practical applications such as content moderation, these are critical weaknesses. Models that misclassify reclaimed slurs penalise the very communities that are commonly targeted by

hate speech. Models that misclassify counter speech undermine positive efforts to fight hate speech. Models that are biased in their target coverage are likely to create and entrench biases in the protections afforded to different groups.

As a suite of functional tests, HateCheck only offers indirect insights into the source of these weaknesses. Poor performance on functional tests can be a consequence of systematic gaps and biases in model training data. It can also indicate a more fundamental inability of the model's architecture to capture relevant linguistic phenomena. **`B-D`** and **`B-F`** share the same architecture but differ in performance on functional tests and in target coverage. This reflects the importance of training data composition, which previous hate speech research has emphasised (Wiegand et al., 2019; Nejadgholi and Kiritchenko, 2020). Future work could investigate the provenance of model weaknesses in more detail, for instance by using test cases from HateCheck to "inoculate" training data (Liu et al., 2019a).

If poor model performance does stem from biased training data, models could be improved through targeted data augmentation (Gardner et al., 2020). HateCheck users could, for instance, sample or construct additional training cases to resemble test cases from functional tests that their model was inaccurate on, bearing in mind that this additional data might introduce other unforeseen biases and limit the validity of HateCheck as a truly external evaluation tool. The models we tested would likely benefit from training on additional cases of negated hate, reclaimed slurs and counter speech. This is supported by findings from Vidgen et al. (2021c), published after HateCheck. They use adversarial data generation to collect these types of challenging cases and train a model that achieves near-perfect accuracy on HateCheck.

3.4 Limitations

3.4.1 Negative Predictive Power

Good performance on a functional test in HateCheck only reveals the absence of a particular weakness, rather than necessarily characterising a generalisable model strength. This *negative predictive power* (Gardner et al., 2020) is common, to some extent, to all finite test sets. Thus,

claims about model quality should not be exaggerated based on positive HateCheck results, and positive results on individual functional tests should be contextualised with results on contrasting functional tests. In model development, HateCheck offers targeted diagnostic insights as a complement to rather than a substitute for evaluation on test sets of real-world hate speech.

3.4.2 Out-Of-Scope Functionalities

Each test case in HateCheck is a separate English-language text document. Thus, HateCheck does not test functionalities related to context outside individual documents, modalities other than text or languages other than English. Future research could expand HateCheck to include functional tests covering such aspects.[10]

Functional tests in HateCheck cover distinct expressions of hate and non-hate. Future work could test more complex compound statements, such as cases combining slurs and profanity.

Further, HateCheck is static and thus does not test functionalities related to language change. This could be addressed by "live" datasets, such as dynamic adversarial benchmarks (Nie et al., 2020; Vidgen et al., 2021c; Kiela et al., 2021).

3.4.3 Limited Coverage

Future research could expand HateCheck to cover additional protected groups, as well as of intersectional characteristics, which interviewees highlighted as a neglected dimension of online hate (e.g. I17: "As a Black woman, I receive abuse that is racialised and gendered").

Similarly, future research could include hateful slurs beyond those covered by HateCheck. This is made easily possible by our template approach to test case generation.

Lastly, future research could craft test cases using more platform- or community-specific language than HateCheck's more general test cases. It could also test hate that is more specific to particular target groups, such as misogynistic tropes.

[10] In work outside of my thesis, we expanded HateCheck to ten additional languages (Röttger et al., 2022b).

3.5 Related Work

Targeted diagnostic datasets like the sets of test cases in HateCheck have been used for model evaluation across a wide range of NLP tasks, such as natural language inference (Naik et al., 2018; McCoy et al., 2019), machine translation (Isabelle et al., 2017; Belinkov and Bisk, 2018) and language modelling (Marvin and Linzen, 2018; Ettinger, 2020). For hate speech detection, however, they have seen very limited use. Palmer et al. (2020) compile three datasets for evaluating model performance on what they call *complex offensive language*, specifically the use of reclaimed slurs, adjective nominalisation and linguistic distancing. They select test cases from other datasets sampled from social media, which introduces substantial disagreement between annotators on labels in their data. Dixon et al. (2018) use templates to generate synthetic sets of toxic and non-toxic cases, which resembles our method for test case creation. They focus primarily on evaluating biases around the use of group identifiers and do not validate the labels in their dataset. Compared to both approaches, HateCheck covers a much larger range of model functionalities, and all test cases, which we generated specifically to fit a given functionality, have clear gold standard labels, which are validated by near-perfect agreement between annotators. Based on HateCheck, Kirk et al. (2022c) created HatemojiCheck, to test functionalities related to emoji-based hate. After HateCheck's publication, Manerba and Tonelli (2021) provided smaller-scale functional tests for abuse detection systems, and Calabrese et al. (2021) devised a hate speech-specific data augmentation technique based on simple heuristics to create additional test cases for a given set of training data, instead of creating a static dataset.

In its use of contrastive cases for model evaluation, HateCheck builds on a long history of minimally-contrastive pairs in NLP (e.g. Levesque et al., 2012; Sennrich, 2017; Glockner et al., 2018; Warstadt et al., 2020). Most relevantly, Kaushik et al. (2020) and Gardner et al. (2020) propose augmenting NLP datasets with contrastive cases for training more generalisable models and enabling more meaningful evaluation. We generated non-hateful contrast cases in our test suite, which is the first application of this kind for hate speech detection.

HateCheck's structure is most directly influenced by the CheckList framework proposed by Ribeiro et al. (2020). However, while they focus on demonstrating the general applicability of

the framework across NLP tasks, we put more emphasis on motivating the selection of functional tests as well as constructing and validating targeted test cases specifically for the task of hate speech detection.

3.6 Conclusion

In this Chapter, we introduced HateCheck, a suite of functional tests for hate speech detection models. We motivated the selection of functional tests through interviews with civil society stakeholders and a review of previous hate speech research, which grounds our approach in both practical and academic applications of hate speech detection models. We designed the functional tests to offer contrasts between hateful and non-hateful content that are challenging to detection models, which enables more accurate evaluation of their true functionalities. For each functional test, we crafted sets of targeted test cases with clear gold standard labels, which we validated through a structured annotation process.

We demonstrated the utility of HateCheck as a diagnostic tool by testing strong academic transformer models as well as two commercial models for hate speech detection. HateCheck showed critical weaknesses for all models that we tested. Specifically, models appeared overly sensitive to particular keywords and phrases, as evidenced by poor performance on tests for reclaimed slurs, counter speech and negated hate. The transformer models we tested also exhibited strong biases in target coverage.

Typically, hate speech detection models have been evaluated using aggregate performance metrics on held-out test data from a small number of widely-used datasets. This is what I denote as Assumption 1: Model Accuracy = Model Quality. HateCheck demonstrates the limits of this simplifying assumption by revealing clear weaknesses in models that are highly accurate on benchmark datasets. In doing so, HateCheck not only contributes to our understanding of models' true qualities and limitations, but also enables targeted model improvements and the development of higher-quality models in the future.

4 Temporal Adaptation of BERT and Performance on Downstream Document Classification: Insights from Social Media

Authors: **Paul Röttger** and **Janet B. Pierrehumbert** (University of Oxford)

Edited Chapter published at **EMNLP 2021** (Findings) as Röttger and Pierrehumbert (2021).

4.1 Introduction

Language use differs between domains and even within a domain, language use changes over time. In different domains, different communities share different social experiences as well as topical interests and thus produce different language (Church and Gale, 1995; Blei et al., 2003). At different times, some topics are discussed more actively while others fade into the background (Church, 2000; Altmann et al., 2009; Pierrehumbert, 2012). Over longer time periods, cultural shifts and linguistic drift add to shorter-term event-driven changes in language use (Hamilton et al., 2016a,b). Training and evaluating NLP models on static datasets disregards these diverse and dynamic properties of language, yet remains common practice. For hate speech detection, I denote this focus on static over dynamic development as Assumption 2: Hate Speech Today = Hate Speech Tomorrow. Static hate speech classifiers perform well on academic data (see Section 2.2), but they cannot account for language change, which is inevitable in practical applications. A particular slur may be used only by some online communities at a certain time, rather than being a general and permanent feature of hate speech. The Covid-19 pandemic, for example, caused a rise in new forms of Anti-East Asian hate (Vidgen et al., 2020). People spreading hate speech may also change their language over time to evade detection (Gillespie, 2018). Model performance, therefore, depends at least in part on how training and test data align in terms of domain and temporality. Nobata et al. (2016) and Florio et al. (2020) found first evidence of this for hate speech detection. Similarly, sentiment analysis

models trained on film reviews perform worse on restaurant reviews (Liu et al., 2019b), and text-based gender and age prediction models trained on one year's data perform increasingly worse on later years (Jaidka et al., 2018).

The widespread use of pre-trained language models like BERT (Devlin et al., 2019) for hate speech detection and other tasks motivates additional considerations about data selection. Such models are first trained *upstream* on large unlabelled corpora to learn general-purpose language representations (*pre-training*) before labelled task data is introduced *downstream* in a separate training phase (*fine-tuning*). In this setting, the choice of unlabelled pre-training data influences downstream model performance like the choice of labelled fine-tuning data does. In particular, we know that *domain* information, i.e. *where* pre-training data is sampled from (e.g. from a particular online platform or community), is highly relevant for downstream tasks. Domain *adaptation*, i.e. additional pre-training of an already-pre-trained model on domain data, has been shown to improve performance on a wide variety of in-domain downstream tasks (Gururangan et al., 2020). By contrast, there is little insight so far into the relevance of *temporality* in pre-training, i.e. *when* pre-training data is sampled from, as it relates to downstream tasks.

In this Chapter, my co-author and I work towards closing this research gap by investigating whether adapting a pre-trained language model to time and domain can improve performance on a downstream document classification task relative to only adapting to domain. Our hypothesis is that temporal adaptation can capture changes in language use such as topical shifts that are relevant to the downstream task, which time-agnostic domain adaptation cannot account for.

First, we attempt to investigate the effects of temporal adaptation for a hate speech detection task, using posts collected from the social media site Gab over ten months. However, we find that the Gab corpus lacks the necessary size and temporal diversity to support robust experimentation. To enable our analysis of temporal adaptation, we therefore create a new benchmark corpus of English-language text comments sampled from the social media site Reddit over three years. The corpus, which we call the Reddit Time Corpus (RTC), consists of a large set of 36m unlabelled comments for adaptation and evaluation on an upstream masked language modelling task (MLM), and a smaller set of 0.9m labelled comments for fine-tuning and evaluation on a downstream five-way document classification task, which we call Political Subreddit Prediction

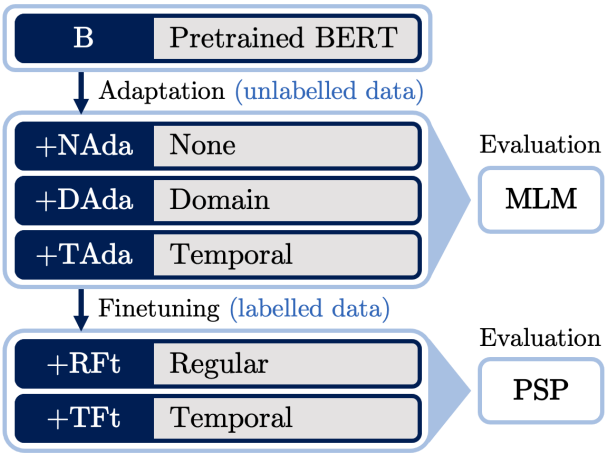

Figure 1: Schematic of our experimental setup. BERT is first adapted in one of three ways using unlabelled data and evaluated upstream on a masked language modelling task (MLM). Either adapted model is then fine-tuned in one of two ways using labelled data and evaluated downstream on Political Subreddit Prediction (PSP), a five-way document classification task.

(PSP). PSP resembles hate speech detection in that it is a document classification task based on social media texts, where we expect linguistic features to differ across online communities and to correlate with political ideology.

We use RTC and a pre-trained BERT model to conduct a series of experiments on the upstream MLM and downstream PSP task (Figure 1). For MLM, we evaluate scale effects in domain adaptation (**DAda**) relative to no adaptation (**NAda**) as well as the effects of temporal adaptation (**TAda**). For PSP, we evaluate scale effects in **DAda** in relation to regular fine-tuning (**RFt**) as well as the effects of temporal fine-tuning (**TFt**). Lastly, we compare PSP performance across all six combinations of these adaptation and fine-tuning strategies (e.g. **TAda+TFt**). Overall, we find that temporal information matters for both tasks. **TAda** improves MLM performance and **TFt** improves PSP performance. **DAda** beats **NAda** on MLM and PSP. However, we do not find clear evidence that **TAda** outperforms **DAda** on PSP. More granular analysis suggests that this is because the event-driven changes in language use captured by **TAda** are not discriminative, i.e. relevant, for the PSP task.[11]

[11]We make our code available on https://github.com/paul-rottger/temporal-adaptation.

4.2 Related Work

Previous work shows that models trained on texts from one time period perform increasingly worse on later time periods for a wide variety of tasks such as review and news article classification (Huang and Paul, 2018, 2019), gender and age prediction (Jaidka et al., 2018), and sentiment analysis (Lukes and Søgaard, 2018). Similar, albeit less comprehensive results also exist for hate speech detection (Nobata et al., 2016; Florio et al., 2020). However, such work has generally not used pre-trained models (e.g. Jaidka et al., 2018) and even if they are used, training and evaluation focuses on labelled task data alone (e.g. Florio et al., 2020). By contrast, our analysis aims to investigate the effects of unsupervised temporal adaptation in pre-training on downstream task performance.

Within the current paradigm of using pre-trained language models, research has focused more on the domain of pre-training data than its temporality. BERT and its variants have been pre-trained from scratch on in-domain data to improve performance on tasks such as hate speech detection (Tran et al., 2020; Caselli et al., 2021), as well as tasks in scientific (Beltagy et al., 2019), clinical (Huang et al., 2019) and legal NLP (Zheng et al., 2021). Further, Gururangan et al. (2020) demonstrate that domain adaptation, a second phase of pre-training on in-domain data, similarly improves performance on various in-domain downstream tasks (see also Alsentzer et al., 2019; Chakrabarty et al., 2019; Lee et al., 2020). We use their approach to domain adaptation as a baseline and extend it to temporality.

Incorporating temporal information in model pre-training has received limited attention. Literature on diachronic embeddings for capturing temporal semantic change (e.g. Hamilton et al., 2016b; Rudolph and Blei, 2018; Tsakalidis and Liakata, 2020) is closely related, but mostly concerned with learning representations across a known time span and investigating their dynamics. Hofmann et al. (2021a) jointly model social and temporal information across time periods using a BERT model, showing that this improves performance on MLM and sentiment analysis. However, they do not evaluate task performance across time periods. By contrast, we adapt BERT to specific time periods with the aim of improving performance on a downstream task located in time. More directly related to our approach, Lazaridou et al. (2021) train autoregressive, left-

to-right transformer models from scratch on unlabelled data sampled up to a specific point in time and then evaluate them on a language modelling task using later data. They find that performance degrades over time and demonstrate that dynamic evaluation (Krause et al., 2019), a form of unsupervised online learning, mitigates this degradation. By contrast, the BERT models we use learn representations through MLM and are adapted to specific time periods, which is less computationally expensive than pre-training from scratch. Most importantly, we go beyond masked language modelling and evaluate the effects of temporal adaptation on a downstream document classification task, which is a more practically relevant use case of pre-trained language models. Following the publication of the original article that this Chapter is based on, several other articles conducted similar experiments evaluating temporal adaptation in relation to downstream tasks such as named entity recognition (Agarwal and Nenkova, 2022), hashtag prediction (Jin et al., 2022) and question answering (Liska et al., 2022).

4.3 Experiments

4.3.1 Initial Experiments for Hate Speech Detection

Our initial plan was to explore the effects of temporal adaptation on a downstream hate speech detection task. For this purpose, we collected text posts from Gab, a Twitter-like social media platform mostly used by the English-speaking far-right. The data we collected, which is publicly available on https://files.pushshift.io/gab/, covers ten months from January to October 2018. For each month, we sampled one million unlabelled posts for model adaptation, plus 10,000 documents for evaluation on a masked language modelling task (MLM). For the same ten months, we also sampled 1,941 posts per month that were labelled for hate speech by Kennedy et al. (2018), with around 8.6% of posts in each month labelled as hateful.

We chose this approach because the labelled Kennedy et al. (2018) dataset provides timestamp metadata, and because it is unique among hate speech datasets in its structured temporal diversity. There is a roughly equal amount of labelled posts for each of the ten months covered by the dataset. Further, unlabelled data, which matches the time span of the labelled data and

is collected from the same platform, is plentiful, readily-available and mostly unmoderated. Collecting equivalent data ex-post on a moderated platform like Twitter would not be possible.

Initial experiments with the Gab data, however, showed that the limited size of the labelled data as well as the strong class imbalance made a robust analysis of the downstream effects of temporal adaptation impossible. The effects of domain and temporal adaptation on upstream MLM with Gab data matched the results we present for Reddit data further below, in that temporal adaptation has a significant positive effect. But the labelled fine-tuning and test sets for each time period would only contain a handful of examples of the hateful class, creating large sampling noise and thus unstable results. These results, which we detail in Appendix B.1, motivated our construction of a new benchmark corpus that covers a larger time span and includes large and balanced labelled subsets for each time period.

4.3.2 Data: Reddit Time Corpus

The Reddit Time Corpus (RTC) covers three years between March 2017 and February 2020 and is split into 36 evenly-sized monthly subsets based on comment timestamps. RTC is sampled from the Pushshift Reddit dataset published by Baumgartner et al. (2020). We provide a data statement (Bender and Friedman, 2018) for RTC in Appendix B.3.

Adaptation: Unlabelled News Comments We collect comments from *r/news* and *r/worldnews*, two of the most subscribed-to and most active subreddits (i.e. discussion forums) on Reddit. *r/news* is primarily focused on US news content while *r/worldnews* describes itself as a "place for major news from around the world, excluding US-internal news". Both subreddits explicitly forbid overtly partisan posts in their community rules. For each of 36 months in our analysis, we sample one million comments, half from each of the two subreddits, for model adaptation. In total, we sample 36 million news comments.

Fine-Tuning: Labelled Politics Comments We collect comments from five subreddits for political discussion: *r/the_donald*, *r/libertarian*, *r/conservative*, *r/politics* and *r/chapotraphouse*. For each of the 36 months in our analysis, we sample 25,000 comments at equal

proportions across these subreddits and label them by the subreddit they were posted to, to create a balanced five-way classification task with equal class distribution across months, which we call Political Subreddit Prediction (PSP). 20,000 comments, sampled with label stratification, are used for model fine-tuning. 5,000 comments are used for evaluation, with labels for PSP and without for MLM. In total, we sample 0.9 million politics comments.

The subreddits we chose for PSP generally correspond to different political ideologies. *r/the_donald* was a subreddit for supporters of then-US President Donald Trump. *r/chapotraphouse* was one of the most active leftist subreddits, which grew out of a popular podcast. Both subreddits were shut down by Reddit in June 2020 for hosting content that promoted hate and violence. *r/conservative* and *r/libertarian* are subreddits for discussing conservative and libertarian politics. *r/politics* is not explicitly ideological but its subscribers tend to be liberal-leaning (Marchal, 2020). We thus expect distinctions between subreddits in PSP to be at least partially predictable based on comment text for two reasons: First, because language use differs between subreddits (Del Tredici and Fernández, 2017). Second, because distinguishing between political subreddits can be seen as a proxy for text-based ideology prediction, which is related to hate speech detection, and a well-established NLP task in its own right (e.g. Conover et al., 2011; Iyyer et al., 2014; Kannangara, 2018; Xiao et al., 2020).

Since both the labelled and unlabelled comments in RTC are sampled from Reddit, we would expect a substantial degree of similarity in language use between them. Based on Jaccard similarity of their vocabularies, comments from different politics subreddits are about as similar to each other as they are to comments from the news subreddits (Table 5). Comments from all subreddits are also more similar to each other than to paragraphs from the BooksCorpus (Zhu et al., 2015) that was used for pre-training BERT, along with English Wikipedia content. This motivates our use of news comments for domain adaptation. Further, we know that topical shifts, particularly those due to exogenous events, can drive changes in language use in both news and politics comments. For instance, Donald Trump's impeachment in December 2019 was immediately and actively discussed in news as well as politics subreddits. This motivates our use of monthly subsets of news comments for adapting models to both domain and time.

	LIB	CTH	CON	POL	T_D	NWN	BC
LIB	1.00	0.42	0.48	0.47	0.44	0.46	0.33
CTH	0.42	1.00	0.43	0.43	0.42	0.42	0.33
CON	0.48	0.43	1.00	0.48	0.46	0.46	0.34
POL	0.47	0.43	0.48	1.00	0.45	0.46	0.34
T_D	0.44	0.42	0.46	0.45	1.00	0.44	0.34
NWN	0.46	0.42	0.46	0.46	0.44	1.00	0.34
BC	0.33	0.33	0.34	0.34	0.34	0.34	1.00

Table 5: Jaccard similarity between vocabularies for random sets of comments (n = 50k) from the five political subreddits (LIB, CTH, CON, POL, T_D) as well as the union of the two news subreddits (NWN) in RTC and a random sample of paragraphs (n = 50k) from the BooksCorpus (BC) used in BERT's pre-training.

PRE-PROCESSING During sampling, we restrict RTC to English-language comments using the `langdetect` Python library.[12] We replace URLs and emojis with [URL] and [EMOJI] tokens, remove line breaks and collapse white space. We remove comments posted by bots, which we identified heuristically.[13] We also remove comments that users have deleted from Reddit and drop duplicates within each monthly subset of the corpus.

4.3.3 MODEL ARCHITECTURE: BERT

We use uncased BERT-base (Devlin et al., 2019) for all experiments. For adapting to unlabelled news comments, we initialise BERT with default pre-trained weights and then continue pre-training on the MLM objective for one epoch, i.e. one pass over all additional data. For fine-tuning on labelled politics comments, we add a linear layer with softmax output and train for three epochs. Further details on model training and parameters as well as model implementation can be found in Appendix B.4.

[12] We found this to work well, but automated language detection is imperfect, especially for short and irregular texts. RTC as a whole still contains some non-English comments, but their relative amount is minimal.

[13] Specifically, we randomly inspected comments and identified those clearly written by auto-moderators and other Reddits bots (e.g. "I AM THE DOG BOT. WOOF WOOF."). We iteratively removed any comments exactly matching their content and format using regex, avoiding false positives, until a subsequent inspection of 100 random comments did not reveal any more comments that were obviously posted by bots. It is likely that RTC as a whole still contains some bot-generated comments, but their relative amount is minimal.

Adaptation Data	Pseudo-Perplexity
0 (= **B+NAda**)	19.54
1 million	8.36
2 million	7.77
5 million	7.10
10 million	6.62

Table 6: Pseudo-perplexity of **B+DAda** on overall MLM test set (n = 5k unlabelled politics comments) for different amounts of adaptation data. Lower pseudo-perplexity is better.

4.3.4 Upstream Task: MLM

Scale Effects in Domain Adaptation First, we evaluate the relative advantage of adapting BERT to domain using unlabelled news comments (**B+DAda**) and the extent to which this advantage scales with the amount of adaptation data. To eliminate temporal effects, news comments for adaptation and evaluation are sampled in equal proportions across all 36 months in RTC. As an evaluation metric, we report pseudo-perplexity, which we calculate as the exponential of the average cross-entropy loss across masked tokens (Table 6).

MLM performance clearly benefits from domain adaptation. Pseudo-perplexity on the test set decreases by 57.22%, from 19.54 to 8.36, for **B+DAda** with one million news comments compared to **B+NAda**. Performance further improves with the amount of adaptation data, but the marginal benefits of more data are clearly decreasing.

Temporal Adaptation Second, we introduce temporality by adapting models to and evaluating them on comments sampled from specific months. We adapt pre-trained BERT to one million news comments from each month in RTC, which yields 36 models (**B+TAda**). We then evaluate each month-adapted model on each monthly test set of 5,000 politics comments, so that in total we perform 1,296 evaluations. Pseudo-perplexity is comparable between models on the same test set but not between different test sets. Thus, we report percentage differences in pseudo-perplexity relative to the pseudo-perplexity of a domain-adapted control model (**B+DAda** with one million news comments) on a given test set. For readability, Table 7 shows results for every fourth month.

Adapt.	17-04	17-08	17-12	18-04	18-08	18-12	19-04	19-08	19-12
17-04	-0.56	0.53	2.10	3.24	4.29	5.37	4.34	5.19	5.74
17-08	0.52	-1.62	1.96	2.89	1.98	4.58	4.05	3.20	4.87
17-12	1.48	0.42	-0.87	1.05	1.50	2.61	2.26	2.65	2.24
18-04	1.14	0.69	1.82	-0.95	0.98	2.47	2.14	2.67	2.54
18-08	1.19	-0.20	-0.06	0.34	-1.12	0.38	1.38	0.81	1.38
18-12	0.79	0.78	0.90	0.69	0.08	-1.03	0.98	1.14	1.34
19-04	2.01	1.09	1.64	1.28	0.18	0.62	-0.66	1.10	-0.19
19-08	3.05	2.06	2.21	1.54	1.85	2.08	1.26	-0.12	1.22
19-12	2.04	1.00	1.47	1.82	1.26	1.23	-0.18	0.70	-2.36

Table 7: % difference in pseudo-perplexity of month-adapted models (**B+TAda**) relative to the control model (**B+DAda**). Rows correspond to adaptation sets (n = 1m unlabelled news comments), columns to test sets (n = 5k unlabelled politics comments).

For each monthly test set of politics comments, the best-performing model is the one adapted to news comments from the same month. When adaptation month matches test month, **TAda** outperforms **DAda** by 1.03% on average. For other months, **TAda** generally performs worse than **DAda**. As the temporal distance between adaptation month and test month increases, **TAda**'s performance decreases. Lastly, relative to the month they were adapted to, models generally perform better on past than on future test data. A one-sided Wilcoxon signed-rank test of all pairs of matching off-diagonal results (e.g. for a model adapted to 17-12, tested on 18-12 compared to a model adapted to 18-12, tested on 17-12) confirms that this finding is highly significant ($p < 0.001$).

TOKEN-LEVEL ANALYSIS To further investigate why **TAda** outperforms **DAda** on MLM when adaptation month matches test month, we analyse changes in cross-entropy loss on individual masked tokens and how they contribute to the overall performance improvement.[14] Since we would expect dynamics of language use to vary between word classes, we use part-of-speech (POS) tags to structure our analysis. Specifically, we apply the `spaCy` POS tagger to all 36 test

[14]BERT uses a WordPiece vocabulary. Each token is an instance of a WordPiece, which is a word or sub-word.

sets, link the tags to the WordPiece tokens generated by BERT and then compare performance improvements on masked WordPiece tokens by POS tag (Figure 2).

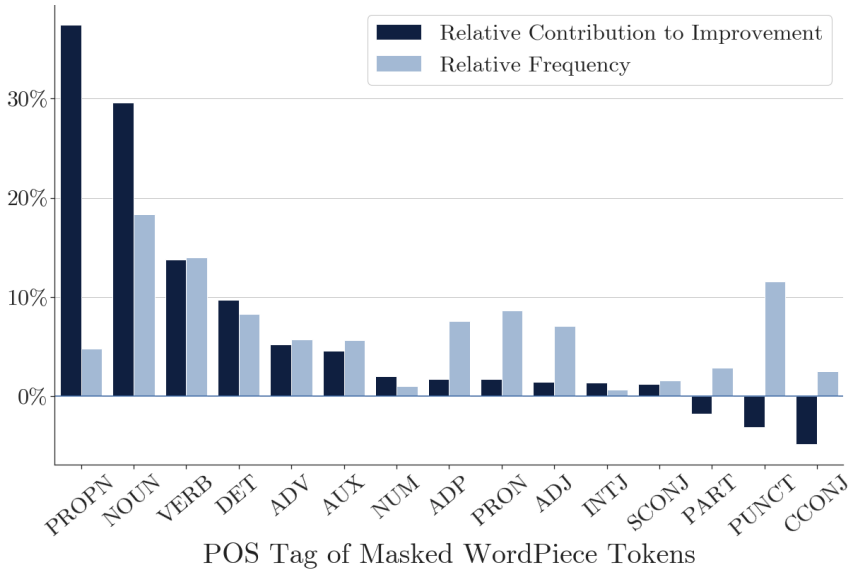

Figure 2: Relative contribution to reduction in cross-entropy loss, **B+TAda** over **B+DAda** on MLM, and relative frequency of masked tokens ($n \approx$ 1m) by POS.

Masked tokens in proper and common nouns drive 67.09% of the overall performance improvement. The former in particular contribute disproportionately much (37.46%) despite making up just 4.76% of all masked tokens. The contribution of tokens in other open-class words like verbs and adverbs roughly matches their frequency. By contrast, tokens in closed-class words such as conjugations contribute disproportionately little.

Proper Noun	**Time**	**Event**
2019-**nC**ov	20-02	WHO Covid press conference
Rex Till**erson**	18-03	Fired by Trump
Aziz Ansari	18-01	Abuse allegations
Kim Foxx	19-04	Prosecuting Jussie Smollet case
Liz **Warren**	19-11	Presidential run
Moscow	19-08	Trump: “Moscow Mitch”
Tide pods	18-02	Meme about eating them
Cville	17-08	“Unite the Right” rally
Ciaram**ella**	19-11	Revealed as CIA whistleblower
Kavana**ugh**	18-10	Supreme Court confirmation

Table 8: Top ten most-improved masked tokens (**bold**) in proper nouns from **TAda** over **DAda**, the test month the tokens are from and the event they correspond to.

Qualitative analysis of those tokens in proper nouns for which cross-entropy loss was reduced the most from **TAda** over **DAda** suggests that **TAda** was most effective in capturing event-driven changes in topical language use (Table 8). These changes are generally bursty. The WordPiece "##ugh" as in "Kavana**ugh**", for example, was not used as part of a proper noun in any 2017 test set. Its use peaked when Kavanaugh was proposed for the US Supreme Court in September 2018 (107/5,000 test comments) and confirmed the month after (67/5,000). After December 2018, it was used at most nine times per test month.

4.3.5 Downstream Task: PSP

Scale Effects in Adaptation and Fine-tuning First, we evaluate relative scale effects in domain adaptation (**DAda**) and regular fine-tuning (**RFt**). To eliminate temporal effects, we sample news comments for adaptation as well as politics comments for fine-tuning and evaluation in equal proportions across all 36 months in RTC. Table 9 reports macro F1 on a scale from 0 to 100. Since there are five balanced classes in PSP, random choice with equal class probability would yield an expected macro F1 of 20.

Adapt.	1k	2k	5k	10k	20k	40k	80k	160k	320k
NAda	34.19	35.70	39.22	41.65	43.22	44.65	46.65	47.92	49.44
1m	37.76	39.26	41.31	42.91	44.03	45.18	46.87	47.94	49.51
2m	38.45	39.57	42.05	42.69	43.43	45.39	46.93	48.38	48.98
5m	38.78	39.84	42.16	43.42	44.46	45.86	47.37	47.93	49.82
10m	38.70	40.37	42.40	43.47	44.53	45.84	47.23	48.44	50.05

Table 9: Macro F1 of **B+DAda+RFt** on overall PSP test set (n = 10k labelled politics comments). Rows correspond to different amounts of adaptation data (unlabelled news comments), columns to different amounts of fine-tuning data (labelled politics comments).

Performance monotonically increases with the amount of politics comments used for **RFt**. Even for large amounts of fine-tuning data, there is no clear sign of a plateau. **DAda** using news comments is relatively more effective when there is less fine-tuning data. Its effectiveness moderately scales with the amount of adaptation data, but the biggest difference in performance is

between the non-adapted model (**NAda**) and the model adapted using one million news comments, particularly for smaller amounts of fine-tuning data.

TEMPORAL FINE-TUNING Second, we introduce temporality to PSP by fine-tuning and evaluating models on labelled politics comments from specific months (**TFt**). We fine-tune a pre-trained, non-adapted BERT model using 20,000 politics comments from each month in RTC, which yields 36 models (**B+NAda+TFt**). We then evaluate each month-tuned model on each monthly test set of 5,000 politics comments. Just like pseudo-perplexity in MLM, macro F1 on PSP is comparable between models on the same test set but not between different test sets. Therefore, we report percentage differences in macro F1 relative to the macro F1 of a control model with regular fine-tuning (**B+NAda+RFt** with 20,000 politics comments) on a given test set. For readability, we report results only for every fourth month in Table 10.

Finetune	17-04	17-08	17-12	18-04	18-08	18-12	19-04	19-08	19-12
17-04	6.81	0.48	-0.23	-1.13	-5.83	-1.32	-4.47	-8.58	-8.32
17-08	-1.20	6.00	-0.32	0.67	-3.15	0.28	-2.77	-5.02	-5.23
17-12	-4.84	1.16	4.95	-2.76	-0.81	-1.03	-5.16	-5.80	-2.63
18-04	-2.76	-1.72	-0.14	3.60	-0.04	0.37	-2.93	-5.14	-6.28
18-08	-2.71	-0.54	-0.66	-0.99	1.35	1.70	-2.89	-3.07	-5.70
18-12	-3.91	-1.64	-3.53	-0.89	-0.70	6.01	-0.79	-2.85	-5.22
19-04	-5.89	-4.82	-4.72	-2.96	-1.51	-0.49	6.12	-0.58	1.57
19-08	-10.82	-6.43	-7.52	-3.62	-4.62	-2.13	-1.10	3.88	-1.78
19-12	-10.31	-7.02	-5.82	-5.35	-4.88	-1.93	-4.75	0.92	3.03

Table 10: % difference in macro F1 of month-tuned models (**B+NAda+TFt**) relative to the control model (**B+NAda+RFt**). Rows correspond to fine-tuning sets (n = 20k labelled politics comments), columns to test sets (n = 5k labelled politics comments).

Overall, the results for month-tuned **TFt** models on PSP (Table 10) resemble those for month-adapted **TAda** models on MLM (Table 7). The best-performing model on a given test month is the one fine-tuned on politics comments from that month. When fine-tuning month matches test month, **TFt** outperforms **RFt** by 5.09% on average. For other months, **TFt** generally performs worse than **RFt**, although there are some exceptions when fine-tuning and test month are not

far apart. For instance, **TFt** models on average perform 1.11% better than the **RFt** model on the test month directly after their fine-tuning month. As temporal distance between fine-tuning and test month grows, the performance of **TFt** models generally worsens. Models generally perform better on past than on future test data relative to the month they were fine-tuned on. A one-sided Wilcoxon signed-rank test of all pairs of matching off-diagonal results confirms that this finding is highly significant ($p < 0.001$).

ADAPTATION AND DOWNSTREAM EFFECTS Third, we compare PSP performance across all six combinations of adaptation (**NAda**, **DAda**, **TAda**) and fine-tuning strategies (**RFt**, **TFt**). Our main interest is in evaluating whether **TAda** provides additional performance benefits on PSP compared to **DAda**. As an evaluation metric, we report average macro F1 across all 36 monthly PSP test sets. For models that incorporate temporality in adaptation (**TAda**) and/or fine-tuning (**TFt**), we consider those where adaptation and/or fine-tuning month matches the test month. Given that we found adaptation to be more effective for smaller amounts of fine-tuning data (Table 9), we report results for fine-tuning sizes of 2,000 and 20,000 (Table 11).

	2k	**20k**
B+NAda+RFt	35.95	43.21
B+DAda+RFt	**39.11**	**43.84**
B+TAda+RFt	39.01	43.81
B+NAda+TFt	37.59	45.41
B+DAda+TFt	40.19	46.02
B+TAda+TFt	**40.38**	**46.12**

Table 11: Average macro F1 across all 36 monthly PSP test sets (n = 180k labelled politics comments) for the six main model configurations, split between models using **RFt** and **TFt**. Best performance is **bold**. Columns correspond to different amounts of labelled politics comments used for fine-tuning.

Our central finding is that models adapted to time and domain (**TAda**) show no clear performance improvement over models adapted to just domain (**DAda**). Macro F1 is marginally higher for **B+TAda+TFt** than **B+DAda+TFt**, but marginally lower for **B+TAda+RFt** than **B+DAda+RFt**. Further, we find that **DAda** outperforms **NAda** and that **DAda** is more beneficial for models fine-tuned on less data, which matches results from Table 9.

MLM IMPROVEMENTS AND PSP PERFORMANCE Finally, we investigate why the benefits of **TAda** over **DAda** on MLM (Table 7) did not manifest in better performance on the downstream PSP task. For this purpose, we focus on masked tokens in proper nouns, which we identified as the main driver of **TAda**'s MLM improvements (Figure 2).

Figure 3 shows that **TAda** improvements in MLM performance on masked tokens in proper nouns are overwhelmingly driven by the top 10% most-improved tokens (e.g. "2019-**nC**ov"), which we found to closely map onto exogenous news events (Table 8). We relate this set of most-improved tokens to PSP as follows: For each token, we take the WordPiece it is an instance of and the PSP test set the comment with the token is from. In that test set, we then count how many subreddits the WordPiece was used in and how many comments in each subreddit the WordPiece was used in, filtering only uses in proper nouns. For example, one of the most-improved tokens is "Kavana**ugh**" in October 2018. That month, the WordPiece "##ugh" was used in a proper noun in 67 out of 5,000 test comments across all five politics subreddits.

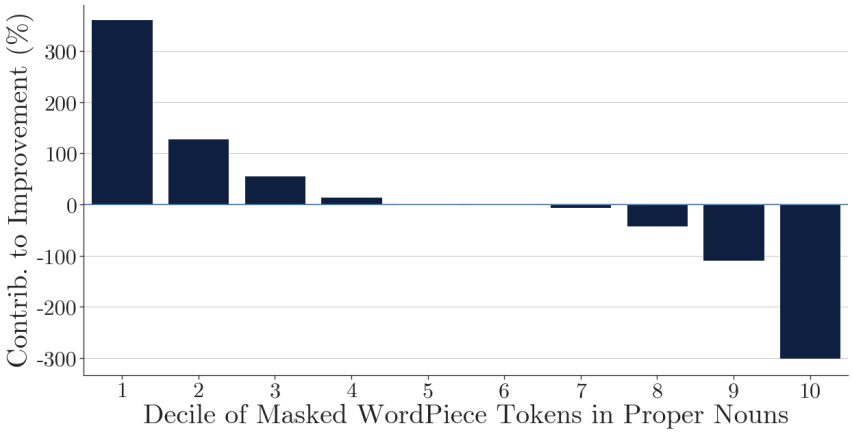

Figure 3: Relative contribution to reduction in cross-entropy loss, **B+TAda** over **B+DAda** on MLM, for masked tokens in proper nouns (n = 48,107) by decile.

Table 12 below suggests that the tokens in proper nouns that drive MLM improvements of **TAda** are not relevant to the PSP task. First, most WordPieces corresponding to these tokens are not distinctive of individual subreddits, with 2,717 WordPieces (65.95%) used in more than one subreddit in a given test set. The more subreddits the WordPieces are used in, the more frequent they are overall and within each subreddit they are used in. Second, more distinctive WordPieces are much rarer. WordPieces that are used in fewer subreddits are used much less

frequently overall and used less frequently in the subreddits that they do appear in. The 1,403 WordPieces (34.05%) that are used in just one subreddit in a given test set are used on average in just 1.28 comments. 1,156 WordPieces are used in just one comment.

Subs.	WordPs	Comments	Avg. Freq.
1	1,403	1,789	1.28
2	769	2,316	1.51
3	559	3,336	1.99
4	570	6,950	3.05
5	819	33,090	8.08

Table 12: Frequency measures for WordPieces corresponding to the top 10% most-improved tokens in proper nouns by **TAda** over **DAda** for MLM (n = 4,120 after deduplication). Grouping is by the number of subreddits (n = 5) that a given WordPiece was used in in a given PSP test set (n = 5k). Average frequency is calculated for the subreddits the WordPieces were used in (Avg. Freq. = Comments / WordP's / Subs.).

4.4 Discussion

4.4.1 Results

We find that **DAda** yields large performance improvements on upstream MLM (Table 6) and the downstream PSP classification task (Table 11) when compared to **NAda**, which matches previous findings (Alsentzer et al., 2019; Chakrabarty et al., 2019; Lee et al., 2020; Gururangan et al., 2020). Further, **DAda** is more effective when there is little fine-tuning data (Table 9).

We also find that temporality matters for both MLM and PSP. For upstream MLM, **TAda** outperforms **DAda** when adaptation month matches test month (Table 7). For downstream PSP, **TFt** outperforms **RFt** when fine-tuning month matches test month (Table 10). For both tasks, model performance decreases as the temporal distance between (pre-)training and test set grows. These findings are consistent with previous evidence for MLM (Lazaridou et al., 2021) and other document classification tasks (e.g. Huang and Paul, 2018; Jaidka et al., 2018; Lukes and Søgaard, 2018), including hate speech detection (Nobata et al., 2016; Florio et al., 2020). The results also illustrate a trade-off between temporal specificity and generalisability across time periods,

which mirrors an equivalent trade-off in domain adaptation (Gururangan et al., 2020). Further, relative to the month they were adapted to (Table 7) or fine-tuned on (Table 10), models perform significantly better on past than on future test sets for MLM and PSP. This matches evidence on the usage of topical words, which tend to occur in bursts, often triggered by an exogenous event, followed by a slower decay (Church, 2000; Altmann et al., 2009; Pierrehumbert, 2012).

Despite these positive results for the individual tasks, we cannot confirm that the benefits of **TAda** over **DAda**, which are evident in MLM, transfer downstream to PSP. **DAda** and **TAda** perform about equally well on PSP (Table 11). This holds for different fine-tuning strategies (**RFt** and **TFt**) and different amounts of fine-tuning data.

Several trivial explanations for this negative finding can be eliminated due to our systematic experimental approach. First, we know that the language used in news comments is informative for PSP, since **DAda** consistently outperforms **NAda** on PSP (Tables 9 and 11). Second, we know that discriminatory language cues for PSP change over time, since **TFt** consistently outperforms **RFt** when fine-tuning month matches test month, and since **TFt** performs increasingly worse as temporal distance between fine-tuning and test month increases (Table 10). Lastly, we know that **TAda**, which uses news comments, allows models to capture some changes in language use in politics comments, since for each monthly test set of politics comments in MLM, the best-performing model is the one adapted to news comments from that same month (Table 7). Therefore, we can conclude that the changes in language use in politics comments that are captured by **TAda** using MLM on news comments are by and large not the changes in discriminatory language cues that are relevant to PSP.

In our token-level analysis, we find that most of **TAda**'s improvements over **DAda** on MLM (Figure 2) are for masked tokens in nouns. Predictions on masked tokens in proper nouns improve disproportionately much, especially for tokens that directly correspond to bursty changes in topical language use driven by exogenous news events (Table 8). However, in relation to the PSP task, the WordPieces corresponding to these tokens generally appear non-discriminative, since most of them are used in several politics subreddits rather than just one (Table 12). Intuitively, many news events are not just relevant to one political ideology, although they may differ in the way they are framed (Card et al., 2015; Demszky et al., 2019; Hofmann et al.,

2021b). In March 2018, for example, when Donald Trump fired his secretary of state Rex Tillerson, an *r/politics* user in the corresponding test set said they were "sympathetic" to him, while an *r/the_donald* user called him a "globalist cuck". Since **TAda** uses comments from news subreddits, it cannot easily capture such distinctive frames.

4.4.2 Promising Uses of Temporal Adaptation

Based on our findings for this particular application of **TAda**, we can formulate positive expectations about the circumstances in which **TAda** would likely be more effective, especially in relation to the task of hate speech detection.

First, we expect **TAda** to be more effective if it captured changes in language use that were more specific to individual classes in the downstream task, i.e. more discriminative. Such changes in language use would occur when an event is relevant to just one class or when the same event is relevant to different classes at different times. For instance, learning about a regional news event in adaptation would likely help a classifier distinguish between comments from regional news sites. Similarly, adapting to the emergence of a new slur which mostly occurs in hateful content, like "Kungflu" and other anti-East Asian slurs that appeared during the Covid-19 pandemic (Vidgen et al., 2020), would likely aid hate speech detection.

Second, we expect **TAda** to be more effective over time scales longer than the 36 months covered by RTC. News and politics are suitable domains for our analysis because topical shifts are visible on short time scales (Figure 2), but over decades and centuries rather than months and years, cultural shifts and linguistic drift add to shorter-term event-driven changes in language use (Hamilton et al., 2016a,b). The primary meaning of the word "gay", for example, shifted from "joyful" in the 1900s towards "homosexual" in the 1990s, and has since remained largely unchanged (Hamilton et al., 2016a). This kind of phenomenon would be highly relevant to hate speech detection operating over long time scales. More generally, we would expect temporal adaptation to improve model performance for any task based on long-term corpora.

Lastly, we may also expect that **TAda** would be more effective if it used pre-training objectives that were more aligned with downstream tasks. Clark et al. (2020a) argue that there is an

inherent mismatch between task-agnostic pre-training that uses masked tokens and fine-tuning that does not, which recent work on discriminative pre-training tries to resolve (Clark et al., 2020b). Future work could explore the use of such techniques for model adaptation.

Even in circumstances in which **TAda** is effective, researchers and practitioners will need to consider the performance trade-off between temporal specificity and generalisability across time periods. For example, when deploying a hate speech detection model for content moderation, performance on newly posted content is more important than performance on historical content, and tailoring the model to the current month is desirable even at a cost to reduced performance on past months. However, for applications where temporality is less relevant, more heterogeneous training data sampled across months is preferable.

4.5 Conclusion

In this Chapter, we investigated whether adapting a pre-trained BERT model to time and domain can increase its performance on a downstream document classification task compared to only adapting it to domain. Overall, we found no clear evidence for this. By devising a systematic experimental approach based on the novel RTC benchmark corpus, we showed that temporality is relevant for both upstream MLM and the downstream PSP document classification task. Temporal adaptation improved MLM performance and temporal fine-tuning improved PSP performance. Further, domain adaptation improved performance on both tasks. Time-specific models generally performed better on past than on future test sets for both tasks, which matches evidence on the bursty usage of topical words. However, the upstream benefits of temporal adaptation for MLM did not translate into better downstream performance on PSP compared to domain adaptation alone. Token-level analysis showed that temporal adaptation captured event-driven changes in language use in downstream task data, but not those changes that are relevant to performance on it. This suggests that temporal adaptation may well be effective for other tasks under circumstances we outlined, which future work could investigate.

Data scarcity prevented us from running our full set of experiments for hate speech detection, which presents a clear avenue for future work that will require larger-scale data collection and

annotation. This will be particularly difficult for hate speech because content moderation on most platforms makes it hard to find such content long after it has been posted. Still, the results we found still speak to the importance of considering the impacts of language change on hate speech detection models. Most commonly, we train and evaluate these models on static datasets. This is what I denote as ASSUMPTION 2: HATE SPEECH TODAY = HATE SPEECH TOMORROW. Our experiments problematise this simplifying assumption, because they clearly show that language on social media changes, over short time scales, in ways that are relevant for both upstream language modelling and downstream classification tasks. While temporal adaptation may not be a silver bullet, temporal fine-tuning, where we train on labelled data from the current time period, showed clear performance benefits in our work and many others (Section 4.2). Conversely, static classification models grow stale. As such, this Chapter further highlights the challenge of language change, as it is relevant to hate speech detection and other classification tasks, and increases our knowledge of how to address it.

5 Two Contrasting Data Annotation Paradigms for Subjective NLP Tasks

Authors: **Paul Röttger** (University of Oxford, Alan Turing Institute), **Bertie Vidgen** (Alan Turing Institute), **Dirk Hovy** (Bocconi University), and **Janet B. Pierrehumbert** (University of Oxford)

Edited Chapter published as a short paper at **NAACL 2022** (Main) as Röttger et al. (2022c).

5.1 Introduction

Many natural language processing (NLP) tasks are subjective, in the sense that there are diverse valid beliefs about what the correct data labels should be. Hate speech detection is a core example of a task that is highly subjective: different people have very different beliefs about what should or should not be labelled as hateful (Talat, 2016; Salminen et al., 2019; Davani et al., 2021), and while some beliefs are more widely accepted than others, or codified in laws, there is no single objective truth. Other examples include toxicity (Sap et al., 2019, 2022), harassment (Al Kuwatly et al., 2020), harmful content (Jiang et al., 2021) and stance detection (Luo et al., 2020; AlDayel and Magdy, 2021) as well as sentiment analysis (Kenyon-Dean et al., 2018; Poria et al., 2020). But even for seemingly objective tasks like part-of-speech tagging, there is subjective disagreement between annotators (Plank et al., 2014b).

Subjective annotator disagreement, also referred to as *human label variation* (Plank, 2022), is often discarded in favour of single majority decisions – a simplification which for hate speech detection I denote as Assumption 3: Hate Speech for Me = Hate Speech for You. In this Chapter, my co-authors and I argue that dataset creators should more actively manage the role of annotator subjectivity in the annotation process, and either explicitly encourage it or discourage it, thus reframing a simplifying assumption that is often left implicit into an active choice. Annotators may subjectively disagree about labels (e.g., for hate speech) but dataset creators can and should decide, based on the intended downstream use of their dataset, whether

they want to a) capture *different beliefs* or b) encode *one specific belief* in their data.

As a framework, we propose two contrasting data annotation paradigms. Each paradigm facilitates a clear and distinct downstream use. The *descriptive* paradigm encourages annotator subjectivity to create datasets as granular surveys of individual beliefs. Descriptive data annotation thus allows for the capturing and modelling of different beliefs. The *prescriptive* paradigm, on the other hand, discourages annotator subjectivity and instead tasks annotators with encoding one specific belief, formulated in the annotation guidelines. Prescriptive data annotation thus enables the training of models that seek to consistently apply one belief. A researcher may, for example, want to model different beliefs about hate speech (→ descriptive paradigm), whereas a content moderation engineer at a social media company would likely want a model to apply their specific content policy (→ prescriptive paradigm).

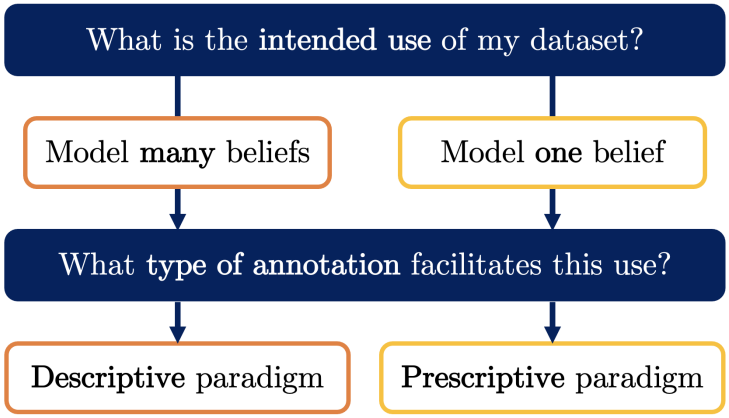

Figure 4: Two key questions for dataset creators.

Neither paradigm is inherently superior, but explicitly aiming for one or the other is beneficial because it makes clear what an annotated dataset can and should be used for. For example, data annotated under the descriptive paradigm can provide insights into different beliefs (Section 5.2.1), but it cannot easily be used to train models with one pre-specified behaviour (Section 5.3.1). By contrast, leaving annotator subjectivity unaddressed, and still aggregating majority labels to simplify model development, as has mostly been the case in NLP so far, leads to datasets that neither capture an interpretable diversity of beliefs nor consistently encode one specific belief. This is an undesirable middle ground without a clear downstream use.[15]

[15]See Appendix C.1 for a selective overview of existing datasets and their alignment with the two paradigms.

Hate speech detection is a particularly fitting application, but the two paradigms are applicable to all data annotation. They can be used to categorise and compare existing datasets, and to make and communicate decisions about how new datasets are annotated. We hope that by naming and explaining the two paradigms, and by discussing key benefits and challenges in their implementation, we can support more intentional annotation process design, which will result in more useful NLP datasets.

TERMINOLOGY Our use of the terms *descriptive* and *prescriptive* aligns with their use in both linguistics and ethics. In linguistics, descriptivism studies how language *is* used, whereas prescriptive grammar declares how language *should* be used (Justice, 2006). Descriptive ethics studies the moral judgments that people make, while prescriptive ethics considers how people ought to act (Thiroux and Krasemann, 2015). Accordingly, descriptive data annotation surveys annotators' beliefs, whereas prescriptive data annotation aims to encode one specific belief, which is formulated in the annotation guidelines.

5.2 THE DESCRIPTIVE ANNOTATION PARADIGM: ENCOURAGING ANNOTATOR SUBJECTIVITY

5.2.1 KEY BENEFITS

INSIGHTS INTO DIVERSE BELIEFS Descriptive data annotation captures a multiplicity of beliefs in data labels, much like a very granular survey would. The distribution of data labels across annotators and examples can therefore provide insights into the beliefs of annotators, or the larger population they may represent. For example, descriptive data annotation has shown that non-Black annotators are more likely to rate African American English as toxic (Sap et al., 2019, 2022). Similar correlations between sociodemographic characteristics of annotators and annotation outcomes have been found in stance (Luo et al., 2020), sentiment (Diaz et al., 2018) and hate speech detection (Talat, 2016).

Even very subjective tasks may have clear-cut entries on which most annotators agree. For

example, crowd workers tend to agree more on the extremes of a hate rating scale (Salminen et al., 2019), and datasets which consist of clear and explicit hate and non-hate can have very high levels of inter-annotator agreement, even with minimal guidelines (Röttger et al., 2021). Conversely, descriptive data annotation can help to identify which entries are more subjective. Jiang et al. (2021), for instance, find that perceptions about the harmfulness of sexually explicit language vary strongly across the eight countries in their sample, whereas support for mass murder or human trafficking is seen as very harmful across all countries.

LEARNING FROM DISAGREEMENT Annotator-level labels from descriptive data annotation have been shown to be a rich source of information for model training. First, they can be used to separately model annotators' beliefs. For less subjective tasks such as question answering, this has served to mitigate undesirable annotator biases (Geva et al., 2019). Davani et al. (2022) reframe and expand on this idea for more subjective tasks like abuse detection, showing that multi-annotator model architectures outperform standard single-label approaches on single label prediction. Second, instead of modelling each annotator separately, other work has grouped them into clusters based on sociodemographic attributes (Al Kuwatly et al., 2020) or polarisation measures derived from annotator labels (Akhtar et al., 2020, 2021), with similar results. Third, models can be trained directly on *soft* labels (i.e., distributions of labels given by annotators), rather than *hard* one-hot ground truth vectors (Plank et al., 2014a; Jamison and Gurevych, 2015; Uma et al., 2020; Fornaciari et al., 2021).

EVALUATING WITH DISAGREEMENT Descriptive data annotation facilitates model evaluation that accounts for different beliefs about how a model should behave (Basile et al., 2021b; Uma et al., 2021). This is particularly relevant when deploying NLP systems for practical tasks such as content moderation, where *user-facing* performance needs to be considered (Gordon et al., 2021). To this end, comparing a model prediction to a descriptive label distribution, the *crowd truth* (Aroyo and Welty, 2015), can help estimate how *acceptable* the prediction would be to users (Alm, 2011).

5.2.2 Key Challenges

Representativeness of Annotators The survey-like benefits of descriptive data annotation correspond to survey-like challenges. First, dataset creators must decide who their data aims to represent, by establishing a clear population of interest. Arora et al. (2020), for example, ask women journalists to annotate harassment targeted at them. Talat (2016) recruits feminist activists as well as crowd workers. Both decisions reflect particular choices about whose beliefs should be represented. Second, dataset creators must consider whether representativeness can practically be achieved. To capture a representative distribution of beliefs for each entry requires dozens, if not hundreds of annotators recruited from the population of interest. Sap et al. (2022), for example, collect toxicity labels from 641 annotators, but only for 15 examples. Other datasets generally use much fewer annotators per entry (see Appendix C.1) and therefore cannot be considered representative in the sense that large (i.e., many-participant) surveys are.

Interpretation of Disagreement In the descriptive paradigm, the absence of a (specified) ground truth label complicates the interpretation of any observed annotator disagreement: it may be due to a genuine difference in beliefs, which is desirable in this paradigm, or due to undesirable annotator error (Pavlick and Kwiatkowski, 2019; Basile et al., 2021a; Leonardelli et al., 2021). The same issue applies to inter-annotator agreement metrics like Fleiss' Kappa. When subjectivity is encouraged, such metrics can at best measure task subjectiveness, but not task difficulty, annotator performance, or dataset quality (Zaenen, 2006; Alm, 2011).

Label Aggregation Descriptive annotation has clear downstream uses (Section 5.2.1) but it is fundamentally misaligned with standard NLP methods that rely on single gold standard labels. When datasets are constructed to be granular surveys of beliefs, reducing those beliefs to a single label, through majority voting or otherwise, goes directly against that purpose. Aggregating labels conceals informative disagreements (Leonardelli et al., 2021; Basile et al., 2021b) and risks discarding minority beliefs (Prabhakaran et al., 2021; Basile et al., 2021a).

5.3 The Prescriptive Annotation Paradigm: Discouraging Annotator Subjectivity

5.3.1 Key Benefits

Specified Model Behaviour Encoding one specific belief in a dataset through data annotation is difficult (Section 5.3.2) but creates clear advantages in many practical applications. Social media companies, for example, moderate content on their platforms according to specific content policies.[16] Therefore, they need data annotated in accordance with those policies to train their content moderation models, or the classifiers that power them. This illustrates that even for highly subjective tasks, where different model behaviours are plausible and valid, one specific behaviour may be practically desirable. Prescriptive data annotation specifies such desired behaviours in datasets for model training and evaluation.

Quality Assurance In the prescriptive paradigm, annotator disagreements are a call to action because they indicate that a) the annotation guidelines were not correctly applied by annotators or b) the guidelines themselves were inadequate. Annotator errors can be found using noise identification techniques (e.g., Hovy et al., 2013; Zhang et al., 2017; Paun et al., 2018; Northcutt et al., 2021), corrected by expert annotators (Vidgen and Derczynski, 2020; Vidgen et al., 2021b) or their impact mitigated by label aggregation. Guidelines which are unclear or incomplete need to be improved by dataset creators, which may require iterative approaches to annotation (Founta et al., 2018; Zeinert et al., 2021). Therefore, quality assurance under the prescriptive paradigm is a laborious but structured process, with inter-annotator agreement as a useful, albeit noisy, measure of dataset quality.

Visibility of Encoded Belief In the prescriptive paradigm, the one belief that annotators are tasked with applying is made visible and explicit in the annotation guidelines. Well-formulated guidelines should give clear instructions on how to decide between different classes,

[16] In March 2021, a whistleblower shared 300-page content guidelines used by Facebook moderators (Hern, 2021).

along with explanations and examples. This creates accountability, in that people can review, challenge and disagree with the formulated belief. Like data statements (Bender and Friedman, 2018), prescriptive annotation guidelines can provide detailed insights into how datasets were created, which can then inform their downstream use.

5.3.2 Key Challenges

Creation of Annotation Guidelines Creating guidelines for prescriptive data annotation is difficult because it requires topical knowledge and familiarity with the data that is to be annotated. Guidelines would ideally provide a clear judgment on every possible entry, but in practice, such perfectly comprehensive guidelines can only be approximated. Further, creating guidelines for prescriptive data annotation requires deciding which one belief to encode in the dataset. This can be a complex process that risks disregarding non-majority beliefs if marginalised people are not included in it (Raji et al., 2020).

Application of Annotation Guidelines Annotators need to be familiar with annotation guidelines to apply them correctly, which may require additional training, especially if guidelines are long and complex. This is reflected in an increasing shift in the literature towards using annotators with task-relevant experience over non-trained crowd workers (e.g. Basile et al., 2019; Röttger et al., 2021; Vidgen et al., 2021b). During annotation, annotators will need to refer back to the guidelines, which requires giving them sufficient time per entry and providing a well-designed annotation interface.

Persistent Subjectivity Annotator subjectivity can be discouraged, but not eliminated. Inevitable gaps in guidelines leave annotators no choice but to apply their personal judgement for some entries, and even when there is explicit guidance, implicit biases may persist. Sap et al. (2019), for example, demonstrate racial biases in hate speech annotation, and show that targeted annotation prompts can reduce these biases but not fully eliminate them. To address this issue, dataset creators should work with a diverse group of annotators, even when annotator subjectivity is discouraged.

5.4 An Illustrative Annotation Experiment

Experimental Design To illustrate the contrast between the two paradigms and their application to hate speech detection, we conducted an annotation experiment. We recruited 60 annotators via Amazon's MTurk crowdsourcing marketplace and randomly assigned them to one of three groups of 20. Each group was given different guidelines to label the same 200 Twitter posts, taken from a corpus annotated for hate speech by Davidson et al. (2017), as either *hateful* or *non-hateful*. All annotators labelled all 200 posts. **G1**, the descriptive group, received a short prompt which directed them to apply their subjective judgement ('Do you personally feel this post is hateful?'). **G2**, the prescriptive group, received a short prompt which discouraged subjectivity ('Does this post meet the criteria for hate speech?'), along with detailed annotation guidelines. **G3**, which we denote as the prescriptive control group, received the prescriptive prompt and a short definition of hate speech but no further guidelines. This is to control for the difference in length and complexity of annotation guidelines between **G1** and **G2**.

Results We evaluate average percentage agreement and Fleiss' κ to measure dataset-level inter-annotator agreement in each group (Table 13). To test for significant differences in agreement between groups, we use confidence intervals computed with a 1000-sample bootstrap. Agreement is very low in the descriptive group **G1** ($\kappa = 0.20$), which suggests that annotators hold varied beliefs about which posts are hateful. However, agreement is significantly higher ($p < 0.001$) in **G2** ($\kappa = 0.78$), which suggests that a prescriptive approach with detailed annotation guidelines can successfully induce annotators to apply a specified belief rather than their subjective view. Further, agreement in the prescriptive control group **G3** ($\kappa = 0.15$) is as low as in descriptive **G1**, which suggests that comprehensive guidelines are instrumental in facilitating high agreement in the prescriptive paradigm. Finally, groups **G1** and **G3** do not differ systematically on which items attract disagreement. This suggests that annotators with little instruction in **G3** tend to apply their subjective views, like **G1**, rather than answering at random. If annotators in **G1** and/or **G3** were answering at random, we would expect low levels of agreement in both groups, but also significant differences in which items attract disagreement.

Group	Avg. % Agree.	Fleiss' κ
G1 - Descriptive	73.90	0.20
G2 - Prescriptive	**93.72**	**0.78**
G3 - Presc. Control	72.50	0.15

Table 13: Inter-annotator agreement metrics for the three groups of 20 annotators on our 200-post binary dataset. Each group labelled the same data using different annotation guidelines. Larger values correspond to more agreement.

REPRODUCIBILITY We provide additional details on our dataset and annotators in a data statement (Bender and Friedman, 2018) in Appendix C.2. Annotation prompts are given in Appendix C.3. Full guidelines, annotated data and code are available at https://github.com/paul-rottger/annotation-paradigms.

5.5 CONCLUSION

In this Chapter, we named and explained two contrasting paradigms for data annotation. The *descriptive* paradigm encourages annotator subjectivity to create datasets as granular surveys of individual beliefs, which can then be analysed and modelled. The *prescriptive* paradigm tasks annotators with encoding one specific belief formulated in the annotation guidelines, to enable the training of models that seek to apply that one belief to unseen data. We argued that dataset creators should explicitly aim for one paradigm or the other to facilitate the intended downstream use of their dataset. We discussed benefits and challenges in implementing both paradigms, and conducted an annotation experiment that illustrates the contrast between them, as well as the high prevalence of subjective disagreements in hate speech detection.

The two paradigms provide a framework for more intentional annotation process design, which helps dataset creators manage and – in the descriptive paradigm – embrace subjectivity rather than discarding it as label noise. As such, the paradigms challenge the use of ASSUMPTION 3: HATE SPEECH FOR ME = HATE SPEECH FOR YOU as a simplifying assumption, reframing it instead as a deliberate choice regarding how to engage with annotator subjectivity, and thus supporting the creation of higher-quality datasets.

6 Data-Efficient Strategies for Expanding Hate Speech Detection into Under-Resourced Languages

Authors: **Paul Röttger** (University of Oxford), **Debora Nozza** (Bocconi University), **Federico Bianchi** (Stanford University) and **Dirk Hovy** (Bocconi University)

Edited Chapter published at **EMNLP 2022** (Main Conference) as Röttger et al. (2022a).

6.1 Introduction

Hate speech is a global phenomenon, but most hate speech datasets so far focus on English-language content (Vidgen and Derczynski, 2020; Poletto et al., 2021). This bias towards English-language data is what I, somewhat pointedly, denote as Assumption 4: Hate Speech in English = All Hate Speech. Working on a shared language enables collaboration, but a lack of non-English resources also hinders the development of effective models for detecting hate speech in other languages. As a consequence, billions of non-English speakers across the world are less protected against online hate, and even the largest social media platforms have clear language gaps in their content moderation systems (Simonite, 2021; Marinescu, 2021).

Zero-shot cross-lingual transfer, where large multilingual language models are fine-tuned on one source language and then applied to another target language, may appear like a potential solution to the issue of language-specific resource scarcity. However, while this approach performs well on some tasks (Conneau et al., 2020; Barbieri et al., 2022), it fails on many others (Lauscher et al., 2020; Hu et al., 2020). For hate speech detection in particular, cross-lingual performance in such so-called zero-shot settings is lacking (Stappen et al., 2020; Leite et al., 2020). For example, zero-shot cross-lingual transfer cannot account for language-specific taboo expressions that play a key role in classification (Nozza, 2021). Conversely, hate speech detection models trained or fine-tuned directly on the target language, i.e. in few- and many-shot

settings, are consistently found to perform best (Aluru et al., 2020; Pelicon et al., 2021).

So, how do we build hate speech detection models for hundreds more languages? We need at least some labelled data in the target language to make models effective, but data annotation is difficult, time-consuming and expensive. It requires resources that are often very limited for non-English languages. Annotating *hateful* content in particular also risks exposing annotators to harm in the process (Vidgen et al., 2019; Kirk et al., 2022a).

In this Chapter, we explore strategies for hate speech detection in under-resourced languages that make *efficient* use of labelled data, to build *effective* models while also minimising annotation cost and risk of harm to annotators. For this purpose, we conduct a series of experiments using mono- and multilingual models fine-tuned on differently-sized random samples of labelled hate speech data in English as well as Arabic, Spanish, Hindi, Portuguese and Italian.

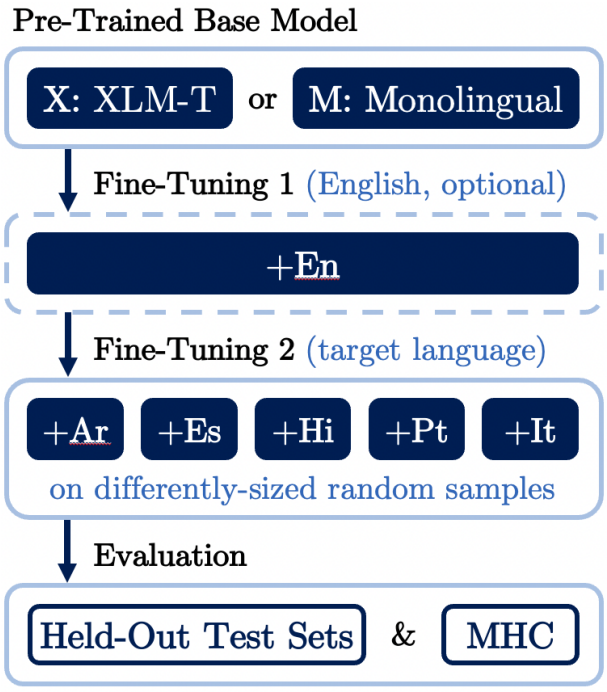

Figure 5: Overview of our experimental setup. We use ISO 639-1 codes to denote the different languages. MHC is Multilingual HateCheck (Röttger et al., 2022b).

Our key findings are:

1. Zero-shot cross-lingual transfer performance is poor. A small amount of labelled target-language data is needed to achieve strong performance on held-out test sets.

2. The benefits of using more such data decrease exponentially.

3. Initial fine-tuning on readily-available English data can partially substitute target-language data and improve model generalisability.

Based on these findings, we formulate and discuss five recommendations for expanding hate speech detection into under-resourced languages:

1. Always collect and label at least a small amount of target-language data.

2. Start by labelling a small set, then iterate.

3. Use diverse data collection methods to increase the marginal benefits of annotation.

4. Use multilingual models for unlocking readily-available data in high-resource languages for initial fine-tuning.

5. Evaluate out-of-domain performance to reveal potential weaknesses in generalisability.

With these recommendations, we hope to facilitate the development of new models for yet-unserved languages, thus helping hate speech research move beyond an overly English language-centric approach to hate speech detection.[17]

DEFINITION OF HATE SPEECH There is no unifying definition of hate speech, and even (quasi-)legal definitions of hate speech vary across cultural and legal settings, as outlined in Section 2.1. Following the prescriptive definition used for HateCheck in Chapter 3, we define hate speech as *abuse that is targeted at a protected group or at its members for being a part of that group*. Protected groups are based on characteristics such as gender identity, race or religion, which broadly reflects Western legal consensus, particularly the US 1964 Civil Rights Act, the UK's 2010 Equality Act and the EU's Charter of Fundamental Rights.[18] Based on these definitions, we approach hate speech detection as the binary classification of content as either hateful or non-hateful.

[17]We make all data and code available at github.com/paul-rottger/efficient-low-resource-hate-detection.
[18]We discuss definitional differences as a limitation of our analysis in Section 6.3.1.

6.2 EXPERIMENTS

All our experiments follow the setup described in Figure 5 further above. We start by loading a pre-trained mono- or multilingual transformer model for sequence classification. For multilingual models, there is an optional first phase of fine-tuning on English data. This is to simulate using readily-available data from a high-resource language that is not the target language. For all models, there is then a second phase of fine-tuning on differently-sized random samples of data in the target language. This is to simulate using scarce data from an under-resourced language. Finally, all models are evaluated on the held-out test set corresponding to the target-language dataset they were fine-tuned on as well as the target-language test suite from Multilingual HateCheck (Röttger et al., 2022b).

6.2.1 DATA

For all our experiments, we use hate speech datasets from hatespeechdata.com, which was first introduced by Vidgen and Derczynski (2020) and is now the largest public repository of datasets annotated for hate, abuse and offensive language. At the time of our review in May 2022, the site listed 53 English datasets as well as 57 datasets in 24 other languages. From these datasets, we select those for our experiments that a) contain explicit labels for hate, and b) use a definition of hate for data annotation that aligns with our own from Section 6.1.

FINE-TUNING 1: ENGLISH For the optional first phase of fine-tuning in English, we use one of three English datasets. DYN21_EN by Vidgen et al. (2021c) contains 41,255 entries, of which 53.9% are labelled as hateful. The entries were hand-crafted by annotators to be challenging to hate speech detection models, using the Dynabench platform (Kiela et al., 2021). FOU18_EN by Founta et al. (2018) contains 99,996 tweets, of which 4.97% are labelled as hateful. KEN20_EN by Kennedy et al. (2020) contains 39,565 comments from Youtube, Twitter and Reddit, of which 29.31% are labelled as hateful.

From each of these three English datasets, we sample 20,000 entries for the optional first phase of fine-tuning, plus another 500 entries for development and 2,000 for testing. To align the

proportion of hate across English datasets, we use random sampling for DYN21_EN, while for KEN20_EN and FOU18_EN we retain all hateful entries and then sample from non-hateful entries. As a result, the proportion of hate in the sampled datasets is 53.9% for DYN21_EN, 50.0% for KEN20_EN and 22.0% for FOU18_EN.

FINE-TUNING 2: TARGET LANGUAGE For fine-tuning models in the target language, we use one of five datasets in five different target languages. BAS19_ES compiled by Basile et al. (2019) for SemEval 2019 contains 4,950 Spanish tweets, of which 41.5% are labelled as hateful. FOR19_PT by Fortuna et al. (2019) contains 5,670 Portuguese tweets, of which 31.5% are labelled as hateful. HAS21_HI compiled by Modha et al. (2021) for HASOC 2021 contains 4,594 Hindi tweets, of which 12.3% are labelled as hateful. OUS19_AR by Ousidhoum et al. (2019) contains 3,353 Arabic tweets, of which 22.5% are labelled as hateful. SAN20_IT compiled by Sanguinetti et al. (2020) for EvalIta 2020 contains 8,100 Italian tweets, of which 41.8% are labelled as hateful.

From each of these five target-language datasets, we randomly sample differently-sized subsets for target-language fine-tuning. Like in English, we set aside 500 entries for development and 2,000 for testing.[19] From the remaining data, we sample subsets in 12 different sizes – 10, 20, 30, 40, 50, 100, 200, 300, 400, 500, 1,000 and 2,000 entries – so that we are able to evaluate the effects of using more or less labelled data within and across different orders of magnitude.[20] Zhao et al. (2021) show that there can be large sampling effects on performance when fine-tuning on small amounts of data. To mitigate this issue, we use 10 different random seeds for each sample size, so that in total we have 120 different samples in each language, and 600 samples across the five non-English languages.

6.2.2 MODELS

MULTILINGUAL MODELS We fine-tune and evaluate XLM-T (Barbieri et al., 2022), an XLM-R model (Conneau et al., 2020) pre-trained on an additional 198 million Twitter posts

[19]Due to limited dataset size, we only set aside 300 dev and 1,000 test entries for OUS19_AR (n=3,353).
[20]There is at least one hateful entry in every sample.

in over 30 languages. XLM-R is a widely-used architecture for multilingual language modelling, which has been shown to achieve near state-of-the-art performance on multilingual hate speech detection (Banerjee et al., 2021; Modha et al., 2021). We chose XLM-T because it strongly outperformed XLM-R across our target language test sets in initial experiments.

MONOLINGUAL MODELS For each of the five target languages, we also fine-tune and evaluate a monolingual transformer model from HuggingFace. For Spanish, we use RoBERTuito (Pérez et al., 2022). For Portuguese, we use BERTimbau (Souza et al., 2020). For Hindi, we use HindiBERT.[21] For Arabic, we use AraBERT v2 (Antoun et al., 2020). For Italian, we use UmBERTo.[22] Details on model training can be found in Appendix D.1.

MODEL NOTATION We denote all models by an additive code. The first part is either M for a monolingual model or X for XLM-T. For XLM-T, the second part of the code is DEN, FEN or KEN, for models fine-tuned on 20,000 entries from DYN21_EN, FOU18_EN or KEN20_EN. For all models, the final part of the code is ES, PT, HI, AR or IT, corresponding to the target language that the model was finetuned on. For example, M+IT denotes the monolingual Italian model, UmBERTo, fine-tuned on SAN20_IT, and X+KEN+AR denotes an XLM-T model fine-tuned first on 20,000 English entries from KEN20_EN and then on OUS19_AR.

6.2.3 EVALUATION SETUP

HELD-OUT TEST SETS + MHC We test all models on the held-out test sets corresponding to their target-language fine-tuning data, to evaluate their in-domain performance (Section 6.2.4). For example, we test X+KEN+IT, which was fine-tuned on SAN20_IT data, on the SAN20_IT test set. Additionally, we test all models on the matching target-language test suite from Multilingual HateCheck (MHC). MHC, published as Röttger et al. (2022b), is an expansion of the functional tests from English HateCheck (Chapter 3) to ten different languages. We use MHC to evaluate out-of-domain generalisability (Section 6.2.5).

[21] https://huggingface.co/neuralspace-reverie/indic-transformers-hi-bert
[22] https://huggingface.co/Musixmatch/umberto-commoncrawl-cased-v1

EVALUATION METRICS We use macro F1 to evaluate model performance because most of our test sets as well as MHC are imbalanced. To give context for interpreting performance, we show baseline model results in all figures: macro F1 for always predicting the hateful class ("always hate"), for never predicting the hateful class ("never hate") and for predicting both classes with equal probability ("50/50"). We also show bootstrapped 95% confidence intervals around the average macro F1, which are calculated across the 10 random seeds for each sample size. These confidence intervals are expected to be wider for models fine-tuned on less data because of larger sampling effects.

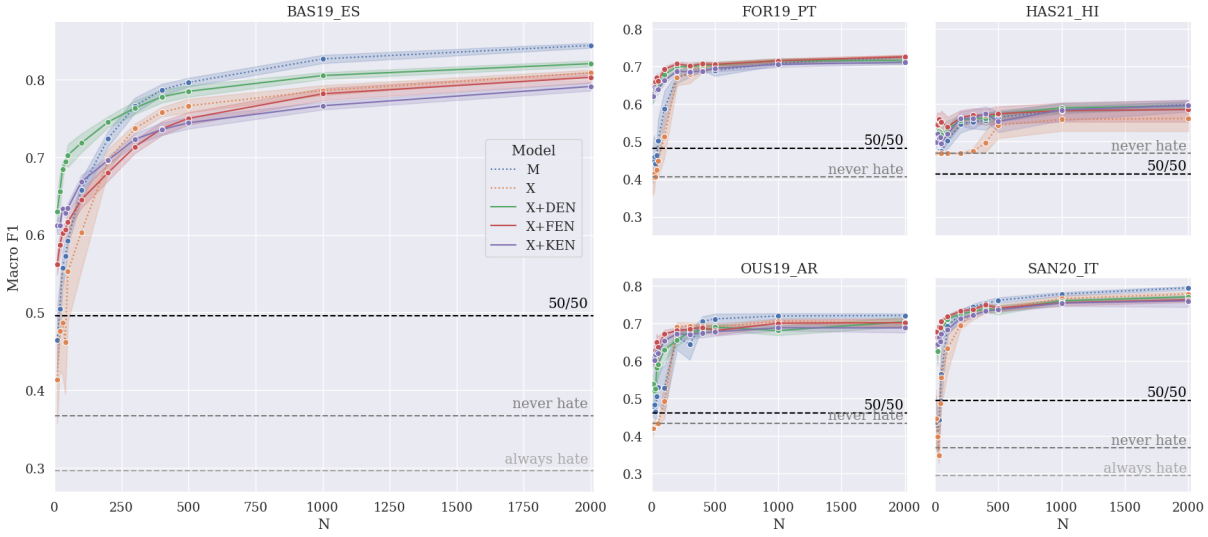

Figure 6: Macro F1 on target-language held-out test sets across models fine-tuned on up to N=2,000 target-language entries. Model notation as described in Section 6.2.2. Confidence intervals and model baselines as described in Section 6.2.4. We provide larger versions of all five graphs in Appendix D.2.

	BAS19_ES			FOR19_PT			HAS21_HI			OUS19_AR			SAN20_IT		
N	**20**	**200**	**2,000**	**20**	**200**	**2,000**	**20**	**200**	**2,000**	**20**	**200**	**2,000**	**20**	**200**	**2,000**
M	0.50	0.72	**0.84**	0.46	0.67	**0.73**	0.47	0.55	**0.60**	0.48	0.66	**0.72**	0.40	**0.73**	**0.79**
X	0.48	0.70	0.81	0.42	0.67	**0.73**	0.47	0.47	0.56	0.43	**0.69**	0.70	0.40	0.70	0.78
X+DEN	**0.66**	**0.75**	0.82	0.63	0.70	0.72	0.52	**0.56**	**0.60**	0.52	0.66	0.70	0.63	**0.73**	0.77
X+FEN	0.59	0.68	0.80	**0.66**	**0.71**	**0.73**	**0.55**	**0.56**	0.59	**0.61**	0.68	0.70	**0.66**	**0.73**	0.76
X+KEN	0.61	0.70	0.79	0.65	0.69	0.71	0.52	**0.56**	**0.60**	0.60	0.67	0.69	0.64	0.71	0.76

Table 14: Macro F1 on respective held-out test sets for models fine-tuned on N target-language entries, averaged across 10 random seeds for each N. Best performance for a given N in **bold**. Results across all N in Appendix D.2.

6.2.4 TESTING ON HELD-OUT TEST SETS

When evaluating our mono- and multilingual models on their corresponding target-language test sets, we find a set of consistent patterns in model performance. We visualise overall performance in Figure 6 and highlight key data points in Table 14, both above.

First, there is an enormous benefit from even very small amounts of target-language fine-tuning data N. Model performance increases sharply across models and held-out test sets up to around N=200. For example, the performance of X+DEN+ES increases from around 0.63 macro F1 at N=10 to 0.70 at N=50, to 0.75 at N=200.

On the other hand, larger amounts of target-language fine-tuning data correspond to much less of an improvement in model performance. Across models and held-out test sets, there is a steep decrease in the marginal benefits of increasing N. M+IT, for example, improves by 0.33 macro F1 from N=20 to N=200, and by just 0.06 from N=200 to N=2,000. X+PT improves by 0.25 macro F1 from N=20 to N=200, and by just 0.06 from N=200 to N=2,000. We analyse these decreasing marginal benefits using linear regression in Section 6.2.6.

Further, there is a clear benefit to a first phase of fine-tuning on English data, when there is limited target-language data. Absolute performance differs across test sets, but multilingual models with initial fine-tuning on English data (i.e. X+DEN, X+FEN and X+KEN) perform substantially better than those without (i.e. M and X), up to around N=200. At N=20, there is up to 0.26 macro F1 difference between the former and the latter. Conversely, models without initial fine-tuning on English data need substantially more target-language data to achieve the same performance. For example, X+DEN+PT fine-tuned on N=100 entries performs as well as M+PT fine-tuned on N=300 entries on the FOR19_PT test set.

Relatedly, there are clear differences in model performance based on which English dataset was used in the first fine-tuning phase, when there is limited target-language data. Among the three English datasets we evaluated, X+DEN performs best for up to around N=200 on the BAS19_ES test set, whereas X+FEN wins out for the other four test sets.

After the strong initial divergence for low amounts of target-language fine-tuning data, performance tends to align across models, regardless of whether they were first fine-tuned on English

data or only fine-tuned on the target language. From around N=200 onwards, for a given test set, all models we evaluate have broadly comparable performance. For example, all models score 0.71 to 0.72 macro F1 on the FOR19_PT test set for N=1,000.

Finally, we find that if there is some divergence in model performance on the held-out test sets for larger amounts of target-language fine-tuning data, then monolingual models tend to outperform multilingual models. For example, despite the disadvantage of monolingual models for N$\leq$200, we find that M+ES achieves 0.84 macro F1 on BAS19_ES for N=2,000, compared 0.82 macro F1 for X+DEN+ES. This difference is visible across test sets and significant at 95% confidence for the Spanish, Arabic and Italian test sets.

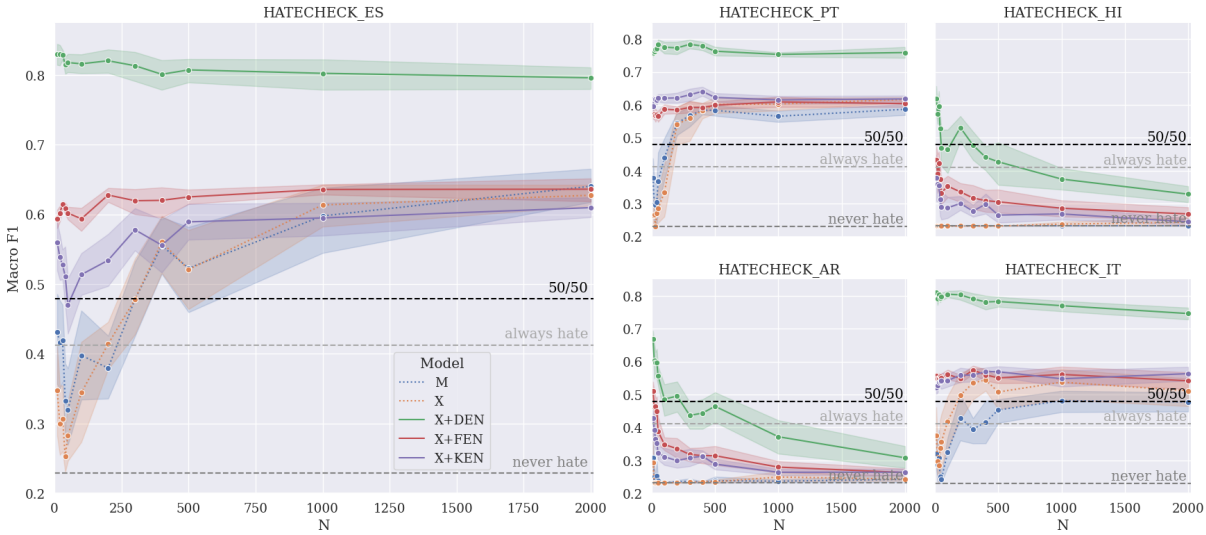

Figure 7: Macro F1 on target-language tests from MHC across models fine-tuned on up to N = 2,000 target-language entries. Model notation as described in Section 6.2.2. Confidence intervals and model baselines as described in Section 6.2.4. We provide larger versions of all five graphs in Appendix D.3.

	HATECHECK_ES			HATECHECK_PT			HATECHECK_HI			HATECHECK_AR			HATECHECK_IT		
N	**20**	**200**	**2,000**	**20**	**200**	**2,000**	**20**	**200**	**2,000**	**20**	**200**	**2,000**	**20**	**200**	**2,000**
M	0.42	0.38	0.64	0.30	0.54	0.59	0.23	0.23	0.23	0.25	0.23	0.24	0.29	0.43	0.48
X	0.30	0.41	0.63	0.27	0.54	0.62	0.23	0.23	0.24	0.23	0.23	0.24	0.30	0.50	0.51
X+DEN	**0.83**	**0.82**	**0.80**	**0.76**	**0.77**	**0.76**	**0.57**	**0.53**	**0.33**	**0.60**	**0.50**	**0.31**	**0.79**	**0.80**	**0.75**
X+FEN	0.60	0.63	0.64	0.57	0.58	0.60	0.39	0.34	0.27	0.46	0.34	0.26	0.55	0.55	0.54
X+KEN	0.54	0.53	0.61	0.62	0.62	0.62	0.36	0.30	0.24	0.39	0.30	0.26	0.53	0.56	0.56

Table 15: Macro F1 on target-language HateCheck for models fine-tuned on N target-language entries, averaged across 10 random seeds for each N. Best performance for a given N in **bold**. Results across all N in Appendix D.3.

6.2.5 TESTING ON MHC

Performance on the target-language test suites from MHC, which none of the models saw during fine-tuning, differs strongly and systematically from results on the held-out test sets (Figure 7). Key data points are highlighted in Table 15 above.

First, as for the held-out test sets, there is an initial benefit to a first phase of fine-tuning on English data, up to around N=500 for MHC. However, there are clear performance differences depending on which English dataset was used. Models fine-tuned on DYN21_EN perform best by far across MHC for any N. This is likely because DYN21_EN contains challenging cases, such as counter speech and reclaimed slurs, which resemble parts of MHC (Vidgen et al., 2021c). On Italian HateCheck for example, X+DEN achieves 0.79 macro F1 at N=20 compared to 0.55 for X+FEN+IT and 0.29 for the monolingual model M+IT.

Second, for Spanish, Portuguese and Italian, there is little to no benefit to fine-tuning on more target-language data, as measured by performance on MHC. Only those models that were not first fine-tuned on English data (i.e. M and X) are improved by fine-tuning on more target-language data, up to around N=500. None of the models benefit from even larger N. On Portuguese HateCheck for example, X+KEN+PT scores 0.62 macro F1 at N=500, N=1,000 and N=2,000.

Third, for Hindi and Arabic, training on more target-language data actively harms performance on MHC. For larger N, all models degrade towards only predicting the non-hateful class. On Arabic HateCheck, for example, X+FEN+AR achieves 0.26 macro F1 at N=2,000, which is only slightly better than the "never hate" baseline at 0.23 macro F1. This is despite the exact same model scoring 0.70 macro F1 on OUS19_AR at N=2,000. We further discuss this finding in Section 6.3.

Lastly, model performance on MHC is much more volatile across random seeds compared to performance on the held-out test sets. This can be seen from the wider confidence intervals in Figure 7 compared to Figure 6.

6.2.6 Regression Analysis

When evaluating models on their corresponding held-out test sets, we saw decreasing benefits to adding more target-language fine-tuning data (Section 6.2.4). To quantify this observation and evidence its significance, we fit ordinary least squares (OLS) linear regressions for each model on each test set. OLS is commonly used in economics and the social sciences to analyse empirical relationships between an outcome variable and one or more regressor variables. In our case, the outcome is model performance as measured by macro F1 and the regressor is the amount of target-language fine-tuning data N. Since the relationship between the two variables is clearly non-linear and plausibly exponential, we regress macro F1 on the natural logarithm of N instead of just N, so that our regression equation for a given random seed i takes the form of

$$F_i = \beta_0 + \beta_1 ln(N_i) + \varepsilon_i$$

where F_i is the observed macro F1 and ε_i the error term. β_0, the intercept, can be interpreted as the expected zero-shot macro F1, i.e. performance at N=0. β_1, the slope, can be interpreted as the expected increase in macro F1 from increasing the amount of target-language fine-tuning data N by 1%, due to our log-linear regression setup. In addition to reporting regression coefficients, we also report adjusted R^2 as a goodness-of-fit measure, which is calculated as the percentage of variance in macro F1 explained by our regression. Lastly, we report the expected percentage-point increase in macro F1 from doubling N, denoted as $2N$, which is calculated as $\beta_1 ln(2)$. Since we saw that variance in macro F1 changes with N, we use heteroskedasticity-robust HC3 standard errors (Hayes and Cai, 2007).

We find that the natural logarithm of the amount of target-language fine-tuning data N provides a strikingly good explanation for model performance as measured by macro F1 (Table 16). R^2 for our regressions, where N is the sole explanatory variable, is generally very high, up to 91.97%, which indicates near-perfect goodness-of-fit. This strongly suggests that there is an exponential decrease in the benefits of using more target-language fine-tuning data. Conversely, there is a constant benefit to doubling the amount of target-language fine-tuning data: We expect the same increase in macro F1 whether we increase N ten-fold from 10 to 20 or from 1,000 to 2,000.

Model: M

Test Set	β_0	β_1	R^2	$2N$
BAS19_ES	0.2879	0.0792	91.97	5.49
FOR19_PT	0.2628	0.0670	71.23	4.65
HAS21_HI	0.2998	0.0607	69.11	4.21
OUS19_AR	0.2092	0.0866	70.08	6.00
SAN20_IT	0.3784	0.0294	77.68	2.04

Model: X+DEN

Test Set	β_0	β_1	R^2	$2N$
BAS19_ES	0.5543	0.0363	87.75	2.52
FOR19_PT	0.5756	0.0205	63.66	1.42
HAS21_HI	0.4691	0.0163	43.91	1.13
OUS19_AR	0.4320	0.0391	64.99	2.71
SAN20_IT	0.5724	0.0272	69.42	1.89

Table 16: OLS results for M and X+DEN. All regression coefficients β_0 and β_1 are significant at $p < 0.001$. We show results for all models in Appendix D.4.

Further, we can confirm our earlier findings that a) initial fine-tuning on English data improves performance for low N, while b) all models perform similarly well for larger N. In our regression, this is visible in monolingual M models having much lower estimated zero-shot performance β_0 paired with much larger β_1 and $2N$ effects compared to the X+DEN models. For example, doubling the amount of Spanish fine-tuning data for M (β_0=0.29) corresponds to an expected increase in macro F1 of 5.49 points on BAS19_ES, whereas for X+DEN (β_0=0.55) we only expect an increase of 2.52 points.

6.3 DISCUSSION AND RECOMMENDATIONS

Based on the sum of our experimental results, we formulate five recommendations for the data-efficient development of effective hate speech detection models for under-resourced languages.

Recommendation 1

Always collect and label at least a small amount of target-language data.

We found enormous benefits from fine-tuning models on even small amounts of target-language data. Zero-shot cross-lingual transfer performs worse than naive baselines, but few shots in the target language go a long way towards improving performance.

Recommendation 2

Do not annotate all the data you collect right away. Instead, start small and iterate.

We found that the benefits of using more target-language fine-tuning data decrease exponentially (Table 16), which suggests that the data in each dataset is not diverse enough to warrant full annotation. The datasets we used in our experiments contained several thousand entries, but most of the useful signal in them can be learned from just a few dozen or hundred randomly-sampled entries.[23] Thus, most of the annotation cost and the potential harm to annotators in compiling these datasets created very little performance benefit. Assuming constant cost per additional annotation, improving F1 by just one percentage point grows exponentially more expensive. Our results suggest that dataset creators should at first annotate smaller subsets of the data they collected and iteratively evaluate what marginal benefits additional annotations are likely to bring compared to their costs.

Recommendation 3

Use diverse data collection strategies to minimise redundant data annotation.

We do not recommend against annotating target-language data. Instead, we recommend against annotating too much data that was collected using the same method. The HAS21_HI dataset, for example, was collected using trending hashtags related to Covid and Indian politics (Modha et al., 2021), and Ousidhoum et al. (2019) used topical keywords related to Islamic inter-sect hate for OUS19_AR. As a consequence, these datasets necessarily have a narrow topical focus as well as limited linguistic diversity. This kind of *selection bias* has been confirmed for sev-

[23] Future work could explore more deliberate selection strategies, like active learning (Markov et al., 2022; Kirk et al., 2022b) or prompt retrieval at inference time (Su et al., 2023), which would likely achieve even better performance with fewer target-language entries. Existing work focuses on monolingual settings.

eral other hate speech datasets (Wiegand et al., 2019; Ousidhoum et al., 2020). MHC, on the other hand, contains diverse statements against seven different target groups in each language (Röttger et al., 2022b). Our results show that very quickly additional entries from a given dataset carry little new and useful information (Figure 6), and that fine-tuning on particularly narrow datasets like HAS21_HI and OUS19_AR can even harm model generalisability as measured by MHC (Figure 7). Conversely, more diverse strategies for data collection, such as sampling from multiple platforms, sampling from a diverse set of users, or using a wider variety of topical keywords, are likely to result in data that is more worth annotating.

Recommendation 4

Use multilingual models to enable the use of readily-available data in high-resource languages for initial fine-tuning.

We found that an initial phase of fine-tuning on English data increases model performance on held-out test sets when there is little target-language fine-tuning data (Figure 6). Thus, English data can partly substitute target-language data in few-shot settings. It also substantially improves out-of-domain generalisability to the MHC test suites (Figure 7), with benefits varying by which English dataset is used. Only multilingual models can access these benefits, because they can be fine-tuned on languages other than just the target language. In our experiments, we used only three of the many English hate speech datasets that have been published to date. Of these, we use only one at a time, and only a subset of the full data. Future work could explore the benefits from initial fine-tuning on larger and more diverse datasets, or a combination of datasets in English and other languages that may be more available than the target language.

Recommendation 5

Do not rely on just one held-out test set to evaluate target language performance.

Monolingual models appear strong when evaluated on held-out test sets, and sometimes even outperform multilingual models when there is relatively much target-language fine-tuning data (Figure 6). However, this hides clear weaknesses in their generalisability as measured by MHC (Figure 7). These findings suggest that monolingual models are more at risk of overfitting

to dataset-specific features than multilingual models fine-tuned first on data from another language. In higher-resource settings, monolingual models are still likely to outperform multilingual models (Martin et al., 2020; Nozza et al., 2020; Armengol-Estapé et al., 2021).

6.3.1 LIMITATIONS

Our analysis is necessarily constrained by the availability of suitable hate speech datasets, particularly for non-English languages. The datasets we used are idiosyncratic, not just because their language differs and their cultural context varies but also because they were collected using different methods at different times. Observed differences in language use, around particular topics or the use of particular types of slurs for example, cannot easily be attributed to one single factor. This is why we focus on analysing general patterns across datasets in different languages, rather than attempting like-for-like comparisons between individual languages. As new datasets are created, future work could expand our analysis to cover more languages as well as more datasets within each language.

Further, our experiments and the recommendations we formulate based on their results presuppose the availability of at least some models tailored to the target language. Multilingual models like XLM-R and XLM-T are respectively pre-trained on 100 and on 30 languages (Conneau et al., 2020; Barbieri et al., 2022), and there are active efforts to expand language model coverage to hundreds of more languages (Wang et al., 2022). However, even the most multilingual models available today cover only a fraction of the world's languages. Similarly, there is a growing number of monolingual transformer models for different languages (Nozza et al., 2020) but most languages are missing such a model.

Also, we focus on varying amounts of fine-tuning data while assuming the existence of held-out development and test sets to evaluate our experiments. However large or small these sets are, labelling them also creates costs and risks of harm to annotators. This needs to be factored into cost assessments for expanding hate speech detection into under-resourced languages.

For reasons of scope and prioritisation, we did not explore the full range of possibilities for improving cross-lingual model performance. For example, we chose English data for the initial

fine-tuning because it is most readily available, but English may not be the optimal choice (Lin et al., 2019). Further, prior work on hate speech detection has found some benefits to data augmentation based on machine translation (Pamungkas et al., 2021; Wang and Banko, 2021) or semi-supervised learning (Zia et al., 2022), which could be explored in future research.

Lastly, our definition of hate speech (Section 6.1) is based on a Western legal consensus, in which hate speech hinges on the target being a protected group. This definition is consistent throughout the datasets we use for our experiments. However, which groups are considered protected may differ across legal and cultural settings. The United Arab Emirates, for example, do not consider sexual orientation a protected characteristic, but do explicitly outlaw "insulting God, his prophets or apostles or holy books or houses of worship or graveyards" (UAE, 2022), which is more restrictive than Western hate speech laws. To avoid Eurocentrism, potential differences across legal and cultural settings need to be accounted for when expanding hate speech detection into new languages and when creating new hate speech datasets.

6.3.2 Ethical Considerations

Language Representation Besides English, our experiments cover a limited set of five languages: Arabic, Spanish, Hindi, Portuguese and Italian. This is due to the very scarcity of non-English datasets for hate speech detection that our work seeks to address. While certainly under-resourced and under-represented in the context of hate speech detection, the five non-English languages we consider are widely spoken, and they have received at least some attention in hate speech research. There is a clear need for creating hate speech datasets in even more under-resourced languages, to facilitate the creation of hate speech detection models in these languages, and to enable even more comprehensive analyses of the kind we conducted here.

Biases in Multilingual Models Zhao et al. (2020) show that multilingual models exhibit similar biases as their monolingual counterparts. Therefore, there is a risk that the models we trained, especially those trained on fewer target-language entries, reflect the biases present in their training and fine-tuning data. Our analyses focused on general model efficacy rather

than bias evaluation. Future work could for example make use of Multilingual HateCheck to evaluate model performance across different targeted groups.

ENVIRONMENTAL IMPACT We fine-tuned 3,000 models for our experiments: five model types for each of five target languages with 12 differently-sized samples for 10 random seeds in each language. However, each model fine-tuning step was very computationally inexpensive, especially because we used very small amounts of fine-tuning data. We were able to run all our experimentation on the University of Oxford's ARC cloud CPUs. Fine-tuning 120 models for a given language took only around four hours for monolingual models and around seven hours for XLM-T. Relative to the concerns raised around the environmental costs of pre-training large language models (Strubell et al., 2019; Henderson et al., 2020; Bender et al., 2021), or even larger-scale fine-tuning with hyperparameter tuning, we therefore consider the environmental costs of our work to be relatively minor. Further, our findings enable the training of more data-efficient models and may thus reduce the environmental impact of future work.

6.4 RELATED WORK

CROSS-LINGUAL TRANSFER LEARNING Only a very small number of the over 7,000 languages in the world are well-represented in natural language processing resources (Joshi et al., 2020). This has motivated much research into cross-lingual transfer, where large multilingual language models are fine-tuned on one (usually high-resource) source language and then applied to another (usually under-resourced) target language. Cross-lingual transfer has been systematically evaluated for a wide range of NLP tasks and for varying amounts of *shots*, i.e. target-language training data. Evidence on the efficacy of *zero-shot* transfer is mixed, across tasks and for transfer between different language pairs (Artetxe and Schwenk, 2019; Pires et al., 2019; Wu and Dredze, 2019; Lin et al., 2019; Hu et al., 2020; Liang et al., 2020; Ruder et al., 2021, inter alia). In comparison, *few-shot* approaches are generally found to be more effective (Lauscher et al., 2020; Hedderich et al., 2020; Zhao et al., 2021; Hung et al., 2022). In *many-shot* settings, multilingual models are often outperformed by their language-specific monolingual counterparts (Martin et al., 2020; Nozza et al., 2020; Armengol-Estapé et al., 2021). In this Chapter,

we confirmed and expanded on key results from this body of prior work, such as the effectiveness of few-shot learning, specifically for the task of hate speech detection. We also introduced OLS linear regression as an effective method for quantifying data efficiency, i.e. dynamic cost-benefit trade-offs in data annotation.

CROSS-LINGUAL HATE SPEECH DETECTION Previous research on cross-lingual hate speech detection generally has a more narrow scope than ours. Nozza et al. (2020) demonstrate weaknesses of zero-shot transfer between English, Italian and Spanish. Leite et al. (2020) find similar results for Brazilian Portuguese, Bigoulaeva et al. (2021) for German, and Stappen et al. (2020) for Spanish, while also providing initial evidence for decreasing returns to fine-tuning on more hate speech data (see also Aluru et al., 2020). Pelicon et al. (2021) and Ahn et al. (2020), like Stappen et al. (2020), find that intermediate training on other languages is effective when little target-language data is available. By comparison, we cover a wider range of languages, with two datasets per language and a consistent definition of hate across datasets. We provide a more systematic evaluation of marginal benefits to additional target-language fine-tuning data based on OLS linear regression, as well as a first assessment of out-of-domain performance using the recently-released Multilingual HateCheck test suites. We are also the first to formulate concrete recommendations for expanding hate speech detection into more under-resourced languages.

6.5 CONCLUSION

In this Chapter, we explored strategies for hate speech detection in under-resourced language settings that are effective while also minimising annotation cost and risk of harm to annotators.

We conducted a series of experiments using mono- and multilingual language models fine-tuned on differently-sized random samples of labelled hate speech data in English as well as Arabic, Spanish, Hindi, Portuguese and Italian. We evaluated all models on held-out test sets as well as out-of-domain target-language test suites from Multilingual HateCheck, and used OLS linear regression to quantify the marginal benefits of target-language fine-tuning data. In our experiments, we found that 1) a small amount of target-language fine-tuning data is needed

to achieve strong performance on held-out test sets, 2) the benefits of using more such data decrease exponentially, and 3) initial fine-tuning on readily-available English data can partially substitute target-language data and improve model generalisability. Based on our findings, we formulated five actionable recommendations for expanding hate speech detection into under-resourced languages.

Hate speech is a global phenomenon, yet most hate speech research and resources so far have focused on English-language content. This is a simplification that I denote as ASSUMPTION 4: HATE SPEECH IN ENGLISH = ALL HATE SPEECH throughout my thesis. Focusing on a shared and widely-spoken language like English enables collaboration, but hate speech detection models are needed in hundreds more languages. With our findings and recommendations, we hope that we can facilitate the development and evaluation of models for yet-unserved languages and thus help to close language gaps that right now leave billions of non-English speakers world less protected against online hate.

7 Conclusion

Summary of Contributions Hate speech detection models play a key role in keeping people safe online. When they work well, they can help find and fight hate at internet scale. And they do work much better today than ever before, in large part due to the adoption of pre-trained language models, which massively improved hate speech detection as well as most other NLP tasks. My thesis, however, showed that these impressive headline results only tell part of the story. Much progress so far has built on simplifying assumptions, which, while useful in some settings, we need to move past in order to develop more truly effective detection models. Continuing to seek accuracy gains on the same widely-used English datasets, for example, will not lead to substantially more robust, more generalisable, or more multilingual models. We need more comprehensive model evaluation to guide more productive research.

In my thesis, I highlighted four simplifying assumptions related to hate speech detection and model evaluation. After defining and discussing core concepts as well as research progress in a literature review (Chapter 2), I presented four Chapters corresponding to four research articles. Each Chapter identified and challenged one specific assumption. Assumption 1 is that Model Accuracy = Model Quality. In Chapter 3, this motivated the creation of HateCheck, a suite of functional tests for hate speech detection models, to allow for more fine-grained diagnostic insights into model quality and reveal critical weaknesses in models which appear accurate on other benchmark datasets. Assumption 2 is that Hate Speech Today = Hate Speech Tomorrow. In Chapter 4, I found that the performance of classification models degrades over time because of language change, and I explored temporal adaptation through continued language model pre-training as a potential solution. Assumption 3 is that Hate Speech for Me = Hate Speech for You. In Chapter 5, I showed that subjectivity has a strong influence on how annotators label hate speech, and I introduced two contrasting data annotation paradigms for managing this subjectivity in hate speech detection and other NLP tasks. Assumption 4 is that Hate Speech in English = All Hate Speech. In Chapter 6, I evidenced the lack of non-English hate speech resources, and explored data-efficient strategies for expanding hate speech detection into more under-resourced languages.

As a whole, my thesis contributes towards better, more comprehensive standards of quality for hate speech detection models. My work identifies concrete issues in current evaluation practices, and introduces novel methods for managing and addressing them. Hate speech detection is the primary focus of my work, but many of my findings generalise to other NLP tasks, which are similarly affected by language change or subjectivity, or focus too much on aggregate performance metrics and English resources. Therefore, I hope that my thesis can be useful and interesting to hate speech researchers as well as a more general NLP audience.

RESEARCH IMPACT Since I published and presented all my thesis research at NLP conferences with a short publication turnaround, my co-authors and I have been fortunate to see the impact of our work even before the end of my DPhil.

The HateCheck suite of functional tests (Chapter 3) has become a community standard for evaluating hate speech detection models. It received substantial media attention because of the poor performance of commercial tools for content moderation that we found.[24] Google Jigsaw, the team behind the Perspective API we tested, is now using HateCheck as part of their model development process. TwoHat, the company behind SiftNinja, used HateCheck to keep testing their tools internally, and then, in 2022, decided to shut down SiftNinja completely.[25] HateCheck is also currently a finalist for Stanford University's AI Audit Challenge.

The temporal adaptation work (Chapter 4), which I presented at EMNLP 2021, was a key reference for recent papers on time-aware language modelling from Google Research (Dhingra et al., 2022), DeepMind (Liska et al., 2022), and other groups (e.g. Jin et al., 2022; Agarwal and Nenkova, 2022; Jang et al., 2022).

Our proposed data annotation paradigms for subjective NLP tasks (Chapter 5) have already been used for a wide variety of NLP tasks (e.g. Demus et al., 2022; Naito et al., 2022; Hallinan et al., 2022), and have informed other work on human variation in data labelling (Plank, 2022).

Our work on data-efficient strategies for expanding hate speech detection into under-resourced languages (Chapter 6) was only published recently, in December 2022.

[24]For example, in the MIT Technology Review, VentureBeat and the Wall Street Journal.
[25]This is based on meetings I had with Google and TwoHat, who reached out after HateCheck was published.

FUTURE RESEARCH Research into hate speech detection, and model evaluation in particular, has made significant progress over the course of my DPhil. Annotator subjectivity in hate speech and other forms of toxic content – and how we can manage and model subjectivity – is now a popular research topic, with its own workshop at LREC 2022, and a dedicated theme at the 2023 ACL Workshop on Online Abuse and Harms.[26] Similarly, non-English resources for hate speech detection are now being created for more and more under-resourced languages. The 2023 AfriHate project, for example, is creating novel datasets in 18 African languages, most of which did not have any dedicated resources so far.[27]

For all the positive trends, however, I also believe that hate speech research still faces some fundamental challenges, which are left unaddressed by my own research and the field as a whole. First, the near-exclusive focus on textual content, and specifically individual text documents from large social media platforms. We know that hate speech in the wild can be multimodal and highly dependent on social and conversational context, yet most research is concerned with classifying individual Twitter posts as hateful or not hateful, in complete isolation. There is some upper bound to what we can learn and discern from text alone, and it is not currently clear how we will move beyond it. Second, and relatedly, the disconnect between academic research into hate speech detection models and the practical realities of content moderation on social media. The field's focus on text-based social media posts is also a consequence of data availability, with the most accessible platforms being the most heavily-researched. By contrast, the platform companies themselves have access to abundant information beyond just text, such as social graphs and other metadata, which they can and do make use of for content moderation. Further, detection models for hate speech and other forms of toxic content are only one component of larger content moderation systems, which also utilise human review, keyword filters and hash lists. Platform companies are generally quite secretive about which models and methods they employ, and what their moderation systems look like. External researchers cannot test these systems, let alone describe them in any detail. Therefore, it is often unclear how advances in academic hate speech research will translate into the kinds of practical impact, like protecting

[26] https://nlperspectives.di.unito.it/ and https://www.workshopononlineabuse.com/.
Disclaimer: I am a co-organiser of the 2023 Workshop on Online Abuse and Harms.
[27] https://github.com/AfriHate/AfriHate. Disclaimer: I am a team member of the AfriHate project.

people and on online platforms, that most often motivate the research in the first place. There is a real risk of an academic ivory tower effect, which calls into question the efficacy of much research in the field right now. Platforms would have to be much more transparent about their content moderation systems to allow for more directly impactful research from academia.

Finally, again on a more positive note, I do believe that hate speech research, and even the current approaches to text-based hate speech detection, still have untapped potential for advancing more general NLP research. As a highly subjective task, hate speech detection could, for example, provide an ideal test bed for exploring social values in large language models. We expect perceptions of hate, and corresponding data labels, to vary between annotators with different values and attributes. If we prompted generative models to complete hate speech tasks, we could use model completions to reason about the implicit values and attributes of the model. Similarly, hate speech data could play a key role in equipping new generations of large language models, which are fine-tuned on large amounts of labelled task data (Bai et al., 2022; Ouyang et al., 2022; Chung et al., 2022), with social values.

OUTLOOK I am very excited to be working in this space at this moment. Language technology promises to make the internet a safer place. Even if these promises still fall short, and technology alone will never be the solution, much progress has already been made, and there are clear gains ahead of us yet. I hope that with my thesis I can contribute to an adjustment in how we evaluate hate speech detection models, and language models more generally, towards better, more comprehensive standards of model quality that account for factors such as subjectivity or language change. Ultimately, it is these standards that guide the field, and I am optimistic that the new standards we are setting now will enable even more meaningful progress in the future.

A Appendices for Chapter 3: HateCheck

A.1 Ethics Statement

This supplementary section addresses relevant ethical considerations that were not explicitly discussed in the main body of the Chapter.

Interview Participant Rights All interviewees gave explicit consent for their participation after being informed in detail about the research use of their responses. In all research output, quotes from interview responses were anonymised. We also did not reveal specific participant demographics or affiliations. Our interview approach was approved by the Alan Turing Institute's Ethics Review Board.

Intellectual Property Rights The test cases in HateCheck were crafted by the authors. As synthetic data, they pose no risk of violating intellectual property rights.

Annotator Compensation We employed a team of ten annotators to validate the quality of the HateCheck dataset. Annotators were compensated at a rate of £16 per hour. The rate was set 50% above the local London living wage at the time (£10.85), although all work was completed remotely. All training time and meetings were paid.

Intended Use HateCheck's intended use is as an evaluative tool for hate speech detection models, providing structured and targeted diagnostic insights into model functionalities. We demonstrated this use of HateCheck in Section 3.3. We also briefly discussed alternative uses of HateCheck, e.g. as a starting point for data augmentation. These uses aim at aiding the development of better hate speech detection models.

Potential Misuse Researchers might overextend claims about the functionalities of their models based on their test performance, which we would consider a misuse of HateCheck. We

directly addressed this concern by highlighting HateCheck's negative predictive power, i.e. the fact that it primarily reveals model weaknesses rather than necessarily characterising generalisable model strengths, as one of its limitations. For the same reason, we emphasised the limits to HateCheck's coverage, e.g. in terms of slurs and identity terms.

A.2 Data Statement

Following Bender and Friedman (2018), we provide a data statement, which documents the generation and provenance of test cases in HateCheck.

A. CURATION RATIONALE In order to construct HateCheck, a first suite of functional tests for hate speech detection models, we generated 3,901 short English-language text documents by hand and by using simple templates for group identifiers and slurs (Section 3.2.4). Each document corresponds to one functional test and a binary gold standard label (hateful or non-hateful). In order to validate the gold standard labels, we trained a team of ten annotators, assigning five of them to each document, and asked them to provide independent labels (Section 3.2.5). To further improve data quality, we also gave annotators the option to flag cases they felt were unrealistic (e.g. nonsensical), but this flag was not used for any one HateCheck case by more than one annotator.

B. LANGUAGE VARIETY HateCheck only covers English-language text documents. We opted for English language since this maximises HateCheck's relevance to previous and current work in hate speech detection, which is mostly concerned with English-language data. Our language choice also reflects the expertise of authors and annotators. We discuss the lack of language variety as a limitation of HateCheck in Section 3.4.2 and suggest expansion to other languages as a priority for future research.

C. SPEAKER DEMOGRAPHICS Since all test cases in HateCheck were hand-crafted, the speakers are the same as the authors. Test cases in the test suite were primarily generated by the lead author, who is a researcher at a UK university. The lead author is not a native

English speaker but has lived in English-speaking countries for more than five years and has extensively engaged with English-language hate speech in previous research. All test cases were also reviewed by two co-authors, both of whom have worked with English-language hate speech data for more than five years and one of whom is a native English speaker from the UK.

D. ANNOTATOR DEMOGRAPHICS We recruited a team of ten annotators to work for two weeks. 30% were male and 70% were female. 60% were 18-29 and 40% were 30-39. 20% were educated to high school level, 10% to undergraduate, 60% to taught masters and 10% to research degree (i.e. PhD). 70% were native English speakers and 30% were non-native but fluent. Annotators had a range of nationalities: 60% were British and 10% each were Polish, Spanish, Argentinian and Irish. Most annotators identified as ethnically White (70%), followed by Middle Eastern (20%) and a mixed ethnic background (10%). Annotators all used social media regularly, and 60% used it more than once per day. All annotators had seen other people targeted by online abuse before, and 80% had been targeted personally.

All annotators had previously completed annotation work on at least one other hate speech dataset. In the first week, we introduced the binary annotation task to them in an onboarding session and tested their understanding on a set of 100 cases, which we then provided individual feedback on. In the second week, we asked each annotator to annotate around 2,000 test cases so that each case in our test suite was annotated by varied sets of exactly five annotators. Throughout the process, we communicated with annotators in real-time over a messaging platform. We also followed guidance for protecting and monitoring annotator well-being provided by Vidgen et al. (2019).

E. SPEECH SITUATION All test cases were created between the 23rd of November and the 13th of December 2020.

F. TEXT CHARACTERISTICS The composition of the dataset, including primary label and secondary labels, is described in detail in Sections 3.2.3 and 3.2.4 of the Chapter.

A.3 References for Functional Tests

F1 – *strong negative emotions explicitly expressed about a protected group or its members*: Resembles "expressed hatred" (Davidson et al., 2017) and "identity attack" (Banko et al., 2020).

F2 – *explicit descriptions of a protected group or its members using very negative attributes*: Refines more general "insult" categories (Davidson et al., 2017; Zampieri et al., 2019a).

F3 – *explicit dehumanisation of a protected group or its members*: Prevalent form of hate (Mendelsohn et al., 2020; Banko et al., 2020; Vidgen et al., 2020). Highlighted in our interviews (e.g. I18: "hate crime [often claims] people are inferior and subhuman.").

F4 – *implicit derogation of a protected group or its members*: Closely resembles "implied bias" (Sap et al., 2020) and "implicit abuse" (Talat et al., 2017; Zhang and Luo, 2019). Highlighted in our interviews (e.g. I16: "hate has always been expressed idiomatically").

F5 – *direct threats against a protected group or its members*: Core element of several hate speech taxonomies (Golbeck et al., 2017; Zampieri et al., 2019a; Vidgen et al., 2020; Banko et al., 2020)

F6 – *threats expressed as normative statements*: Highlighted by an interviewee as a way of avoiding legal consequences to hate speech (I1: "[normative threats] are extremely hateful, but [legally] okay").

F7 – *hate expressed using slurs*: Prevalent way of expressing hate (Palmer et al., 2020; Banko et al., 2020; Kurrek et al., 2020).

F8 – *non-hateful homonyms of slur*: Relevant alternative use of slurs (Kurrek et al., 2020).

F9 – *use of reclaimed slurs*: Likely source of classification error (Palmer et al., 2020). Highlighted in our interviews (e.g. I7: "A lot of LGBT people use slurs to identify themselves, like reclaim the word queer, and people [...] report that and then that will get hidden").

F10 – *hate expressed using profanity*: Refines more general "insult" categories (Davidson et al., 2017; Zampieri et al., 2019a).

F11 – *non-hateful uses of profanity*: Oversensitiveness of hate speech detection models to profanity (Davidson et al., 2017; Malmasi and Zampieri, 2018; van Aken et al., 2018).

F12 – *hate expressed through pronoun reference in subsequent clauses*: Syntactic relationships and long-range dependencies as model weak points (Burnap and Williams, 2015; Vidgen et al., 2019).

F13 – *hate expressed through pronoun reference in subsequent sentences*: See F12.

F14 – *hate expressed using negated positive statements*: Negation as an effective adversary for hate speech detection models (Hosseini et al., 2017; Dinan et al., 2019).

F15 – *non-hate expressed using negated hateful statements*: See F14.

F16 – *hate phrased as a question*: Likely source of classification error (van Aken et al., 2018).

F17 – *hate phrased as an opinion*: Highlighted by an interviewee as a way of avoiding legal consequences to hate speech (I1: "If you start a sentence by saying 'I think that' [...], the limits of what you can say are much bigger").

F18 – *neutral statements using protected group identifiers*: Oversensitiveness of hate speech detection models to terms such as "black" and "gay" (Dixon et al., 2018; Park et al., 2018; Kennedy et al., 2020). Also highlighted in our interviews (e.g. I7: "I have seen the algorithm get it wrong, if someone's saying something like 'I'm so gay'.").

F19 – *positive statements using protected group identifiers*: See F18.

F20 – *denouncements of hate that quote it*: Counter speech as a source of classification error (Warner and Hirschberg, 2012; van Aken et al., 2018; Vidgen et al., 2020). Most mentioned concern in our interviews (e.g. I4: "people will be quoting someone, calling that person out [...] but that will get picked up by the system").

F21 – *denouncements of hate that make direct reference to it*: See F20.

F22 – *abuse targeted at objects*: Distinct from hate speech since it targets out-of-scope entities (Wulczyn et al., 2017; Zampieri et al., 2019a).

F23 – *abuse targeted at individuals not referencing membership in a protected group*: See F22.

F24 – *abuse targeted at non-protected groups (e.g. professions)*: See F22.

F25 – *swaps of adjacent characters*: Simple misspellings can be challenging for detection models (van Aken et al., 2018; Qian et al., 2018). Particularly relevant to hate speech since

they can reflect intentional behaviour of users looking to avoid detection (Hosseini et al., 2017; Gröndahl et al., 2018; Vidgen et al., 2019).

F26 – *missing characters*: Highlighted in our interviews (e.g. I7: "it could be a misspelling of a word like 'f*ggot', and someone's put one 'g' instead of two").

F27 – *missing word boundaries*: Effective adversary for a hate speech detection model (Gröndahl et al., 2018). Resembles the use of hashtags on social media (I2: "there have been a highly Islamophobic hashtags going around").

F28 – *added spaces between characters*: Effective adversary for a hate speech detection model (Gröndahl et al., 2018). Highlighted in our interviews (e.g. I5: "misspellings, missing letters or additional spaces between the letters.").

F29 – *leet speak*: Resembles "obfuscations" (Nobata et al., 2016; van Aken et al., 2018). Highlighted in our interviews (e.g. I14: "[hate speakers] replace letters with numbers").

A.4 Hateful Slurs in HateCheck

For each of the seven protected groups covered by HateCheck, we searched hatebase.org, a crowd-sourced hate speech lexicon, for slurs which target that group. From these slurs, we selected the three that were most often logged by users of the site (e.g. "wh*re", "b*tch" and "sl*t" for women), except for when the third-most sighted slur was logged substantially less often than the second, in which case we selected the top two (e.g. "tr*nny" and "sh*male" for trans people). For immigration status, which is not a target category on hatebase.org, we chose "r*pefugee", a slur for refugees used by the European far right, and "w*tback", a slur for Mexican immigrants to the US, which was logged similarly often as other slurs in HateCheck.

For reclaimed slurs (**F9**), we focus on slurs reclaimed by black communities (particularly "n*gga"), gay communities ("f*g", "f*ggot", "q*eer") and by women ("b*tch"), reflecting the concerns highlighted by our interview participants (e.g. I4: "n*gga would often get [wrongly] picked up by [moderation] systems"). Ahead of the structured annotation process (§3.2.5) and only for test cases with reclaimed slurs, we asked self-identifying members of the relevant

groups in our personal networks whether they would consider the test cases to contain valid and realistic reclaimed slur uses, which held true for all test cases.

Protected Group	**Slurs**
Women	b*tch, sl*t, wh*re
Trans people	tr*nny, sh*male
Gay people	f*ggot, f*g, q*eer
Black people	n*gger, c*on
Disabled people	r*tard, cr*pple, m*ng
Muslims	m*zzie, J*hadi, camel f*cker
Immigrants	w*tbacks, r*pefugees

Table 17: Hateful slurs in HateCheck

A.5 Datasets for Fine-Tuning

A.5.1 Davidson et al. (2017) Data

Sampling Davidson et al. (2017) searched Twitter for tweets containing keywords from a list they compiled from hatebase.org, which yielded a sample of tweets from 33,458 users. They then randomly sampled 25,000 tweets from all tweets of these users.

Annotation The authors hired crowd workers from CrowdFlower to annotate each tweet as *hateful*, *offensive* or *neither*. 92.0% of tweets were annotated by three crowd workers, the remainder by at least four and up to nine. For inter-annotator agreement, the authors report a "CrowdFlower score" of 92%.

Data We used 24,783 annotated tweets made available by the authors on github.com/t-davidson/hate-speech-and-offensive-language. 1,430 tweets (5.8%) are labelled *hateful*, 19,190 (77.4%) *offensive* and 4,163 (16.8%) *neither*. We collapse the latter two labels into a single non-hateful label to match HateCheck's binary format, resulting in 1,430 tweets (5.8%) labelled *hateful* and 23,353 (94.2%) labelled *non-hateful*.

DEFINITION OF HATE SPEECH "Language that is used to expresses hatred towards a targeted group or is intended to be derogatory, to humiliate, or to insult the members of the group".

A.5.2 FOUNTA ET AL. (2018) DATA

SAMPLING Founta et al. (2018) initially collected a random set of 32 million tweets from Twitter. They then used a boosted random sampling procedure based on negative sentiment and occurrence of offensive words as selected from hatebase.org to augment a random subset of this initial sample with tweets they expected to be more likely to be hateful or abusive.

ANNOTATION The authors hired crowd workers from CrowdFlower to annotate each tweet as *hateful*, *abusive*, *spam* or *normal*. All tweets were annotated by five crowd workers. For inter-annotator agreement, the authors report that 55.9% of tweets had four out of five annotators agreeing on a label.

DATA The authors provided us access to the full text versions of 99,996 annotated tweets. These correspond to the tweet IDs made available by the authors on GitHub.[28] 4,965 tweets (5.0%) are labelled *hateful*, 27,150 (27.2%) *abusive*, 14,030 (14.0%) *spam* and 53,851 (53.9%) *normal*. We collapse the latter three labels into a single non-hateful label to match HateCheck's binary format, resulting in 4,965 tweets (5.0%) labelled *hateful* and 95,031 tweets (95.0%) labelled *non-hateful*.

DEFINITION OF HATE SPEECH "Language used to express hatred towards a targeted individual or group, or is intended to be derogatory, to humiliate, or to insult the members of the group, on the basis of attributes such as race, religion, ethnic origin, sexual orientation, disability, or gender".

[28] https://github.com/ENCASEH2020/hatespeech-twitter

A.5.3 Pre-Processing

Before using the datasets for fine-tuning, we lowercase all text and remove newline and tab characters. We replace URLs, user mentions and emojis with [URL], [USER] and [EMOJI] tokens. We also split hashtags into separate tokens using the `wordsegment` Python package.

A.6 Details on Transformer Models

Model Architecture We implemented uncased BERT-base models (Devlin et al., 2019) using the `transformers` Python library (Wolf et al., 2020). Uncased BERT-base, which is trained on lower-cased English text, has 12 layers, a hidden layer size of 768, 12 attention heads and a total of 110 million parameters. For sequence classification, we added a linear layer with softmax output.

Fine-Tuning **`B-D`** was fine-tuned on binary Davidson et al. (2017) data and **`B-F`** on binary Founta et al. (2018) data. For both datasets, we used a stratified 80/10/10 train/dev/test split. Models were trained for three epochs each. Training batch size was 16. We used cross-entropy loss with class weights emphasising the hateful minority class. Weights were set to the relative proportion of the other class in the training data, meaning that for a 1:9 hateful:non-hateful case split, loss on hateful cases would be multiplied by 9. The optimiser was AdamW (Loshchilov and Hutter, 2019) with a 5e-5 learning rate and a 0.01 weight decay. For regularisation, we set a 10% dropout probability.

Hyperparameter Tuning The number of fine-tuning epochs, the learning rate and the training batch size were determined by exhaustive grid search. We used the range of possible values recommended by Devlin et al. (2019): [2, 3, 4] for epochs, [2e-5, 3e-5, 5e-5] for learning rate and [16, 32] for batch size. There were 18 training/evaluation runs for each model. The best configuration was selected based on loss on the 10% development set.

HELD-OUT PERFORMANCE Micro/macro F1 scores on the held-out test sets corresponding to their training data are 91.5/70.8 for **B-D** (Davidson et al., 2017) and 92.9/70.3 for **B-F** (Founta et al., 2018).

COMPUTATION We ran all computations on a Microsoft Azure "Standard_NC24" server equipped with two NVIDIA Tesla K80 GPU cards. The average wall time for each hyperparameter tuning trial of **B-D** was around 17 minutes, and for **B-F** around 70 minutes.

SOURCE CODE Our code is available on github.com/paul-rottger/hatecheck-experiments.

A.7 COMPARISON TO STATE-OF-THE-ART RESULTS

Most previous work that trains and evaluates models on Davidson et al. (2017) and Founta et al. (2018) data uses their original multiclass label format. In the multiclass case, the relative size of the hateful class compared to the non-hateful classes is larger than in the binary case, which is likely why most models do not use class weights. For comparability, we thus fine-tuned unweighted multiclass versions of **B-D** and **B-F**, using the same model parameters described in Appendix A.6.

On multiclass Davidson et al. (2017) data, Mozafari et al. (2019) report a weighted-average F1 score of 91 for their BERT-base model and 92 for BERT-base combined with a CNN. Cao et al. (2020) report a micro F1 of 89.9 for their ensemble-like "DeepHate" classifier. Our unweighted multiclass BERT-base model achieves 90.7 weighted-average F1 and 91.1 micro F1.

On multiclass Founta et al. (2018) data, Cao et al. (2020) report a micro F1 of 79.1 for "DeepHate". Our unweighted multiclass BERT-base model achieves 81.7 micro F1.

Tran et al. (2020) recently achieved SOTA on several other hate speech datasets with their HABERTOR model. They also find that BERT-base consistently performs very near their SOTA. However, they do not evaluate their models on Davidson et al. (2017) or Founta et al. (2018) data.

B Appendices for Chapter 4: Temporal Adaptation

B.1 Initial Experiments for Hate Speech Detection

Data We collected Gab posts for ten months from January to October 2018, which are publicly available on https://files.pushshift.io/gab/. For each month, we sampled one million unlabelled posts for model adaptation, plus 10,000 documents for evaluation on a masked language modelling task (MLM). We also sampled 1,941 posts per month labelled for hate speech by Kennedy et al. (2018), with around 8.6% of posts in each month labelled as hateful.

Model Setup We use the same BERT architecture as for our experiments with RTC, which we describe in Section 4.3.3. We also use the same experimental setups for the upstream masked language modelling and downstream document classification, i.e. hate speech detection task.

Upstream Task: Masked Language Modelling The results for MLM on Gab data match those we found for Reddit data in the main body of the thesis (Section 4.3.4). Domain adaptation drastically increased model performance on the MLM task relative to a non-adapted BERT model, and this performance increase scaled with the amount of adaptation data, although with diminishing returns (Table 18).

Adaptation Data	**Pseudo-Perplexity**
0 ($\rightarrow$ no adaptation)	36.15
1 million	11.12
2 million	10.14
5 million	8.94
10 million	7.91

Table 18: Pseudo-perplexity on overall Gab MLM test set (n = 10,000) for different amounts of adaptation data from Gab.

Further, models adapted to data from specific months performed best on those months, outperforming a control baseline by 5.68% on average, and performance decreased with temporal distance between adaptation and test month (Table 19).

Adapt.	18-01	18-02	18-03	18-04	18-05	18-06	18-07	18-08	18-09	18-10
18-01	-5.76	2.48	5.30	6.22	8.90	15.89	9.20	17.98	15.95	17.57
18-02	-0.21	-5.53	1.15	4.53	7.01	15.39	8.71	16.07	15.58	16.85
18-03	2.40	0.41	-6.52	1.18	5.28	13.49	6.88	14.22	14.28	14.76
18-04	3.17	2.89	-0.32	-7.17	1.61	10.83	5.99	13.15	12.51	13.85
18-05	3.82	3.46	1.57	0.04	-5.22	8.35	3.74	11.97	11.39	12.58
18-06	5.23	6.23	4.50	3.59	1.62	-0.85	0.66	11.02	10.80	11.61
18-07	6.44	6.88	6.24	5.95	5.08	8.04	-5.73	8.44	8.20	9.56
18-08	8.00	8.19	7.16	7.23	6.55	11.59	1.81	-2.34	0.96	5.43
18-09	9.54	9.12	7.93	8.94	9.22	14.23	4.21	5.33	-10.20	0.23
18-10	9.41	10.13	9.48	9.62	9.64	15.06	5.65	8.57	-1.29	-7.45

Table 19: % difference in pseudo-perplexity of month-adapted models relative to the control model across monthly MLM test sets. Rows correspond to adaptation sets (n = 1m), columns to test sets (n = 10k).

DOWNSTREAM TASK: HATE SPEECH DETECTION The results we found for the downstream hate speech detection task speak to the limitations of the Gab corpus and motivate our construction of RTC in the main body of the thesis (Section 4.3.2). Specifically, the performance of models fine-tuned and evaluated on individual months in the Gab corpus is extremely erratic (Table 20). This is likely a consequence of the small number of labelled entries available per month, which, compounded by the strong class imbalance, introduces strong sampling effects. The monthly test sets, for example, contain just 389 entries, of which on average 33 are labelled as hate speech. To address these issues, RTC contains large amounts of labelled data with balanced class labels for each month.

Finetune	18-01	18-02	18-03	18-04	18-05	18-06	18-07	18-08	18-09	18-10
18-01	6.24	-6.72	-12.39	-4.77	3.02	-6.42	1.43	3.70	1.68	-4.77
18-02	3.92	-0.31	5.40	-0.06	11.11	4.51	6.25	8.40	11.80	-1.57
18-03	0.83	1.77	-3.06	-11.23	15.08	3.48	-4.81	11.36	9.18	-3.33
18-04	7.00	5.73	5.40	4.28	13.11	6.00	5.31	11.36	1.77	-2.74
18-05	3.79	1.04	3.07	-1.88	14.72	1.21	6.51	6.34	5.89	5.32
18-06	7.73	1.75	-1.28	-2.64	11.65	1.21	-0.68	0.39	6.50	1.90
18-07	1.53	-6.52	-12.30	0.67	10.06	1.44	7.75	4.98	4.15	-3.31
18-08	5.22	1.75	-3.62	0.00	13.11	4.26	-3.80	15.29	5.29	-2.66
18-09	-0.70	-0.19	-0.21	9.20	11.11	-0.10	3.24	3.70	14.79	-6.31
18-10	11.60	-0.96	1.52	8.86	16.22	8.69	6.69	8.14	16.15	-5.75

Table 20: % difference in macro F1 of month-tuned non-adapted models relative to the control model across monthly HSD test sets. Rows correspond to finetuning sets (n = 1,552), columns to test sets (n = 389).

B.2 Ethics Statement

Data Collection All data in RTC is sampled from the Pushshift Reddit dataset made publicly available by Baumgartner et al. (2020). This dataset, in turn, was collected via Reddit's own public API in line with the site's terms of service. Our use of this Reddit dataset was also approved by the University of Oxford's Central University Research Ethics Committee. Labels for politics comments in RTC were created from comment metadata, so that no manual annotation was necessary.

Data Characteristics We describe the characteristics of RTC in the main body of this paper and provide additional detail in a data statement (Bender and Friedman, 2018) in Appendix B.3. In particular, we highlight RTC's limited scope in terms of data source (Reddit) and language (English), which limits the generalisability of models trained on it.

Intended Use The intended use of temporal adaptation is as an alternative to existing strategies for continued pre-training, particularly domain adaptation. Our article explores a specific application of temporal adaptation using monthly sets of news comments for a downstream

classification task of politics comments. Temporal adaptation could be applied to most other NLP tasks as long as pre-training and task data can be located in time, although the effectiveness of temporal adaptation may differ. Effective temporal adaptation stands to improve diachronic model performance and thus reduce error rates in real-world applications.

POTENTIAL MISUSE As with domain adaptation, temporal adaptation creates a trade-off between specificity and generalisability. Models adapted to a particular time period and domain should not be used for other time periods and domains without careful consideration of resulting biases.

ENVIRONMENTAL IMPACT Temporal adaptation is more computationally expensive than just fine-tuning using a (smaller) set of labelled task data but much less computationally expensive than pre-training from scratch on even larger unlabelled datasets. Relative to the concerns raised around the environmental costs of the latter (Strubell et al., 2019; Henderson et al., 2020; Bender et al., 2021), we consider the environmental costs of temporal adaptation to be relatively minor. In practical applications, researchers could consider cumulative approaches to temporal adaptation, rather than adapting separate models for each time period, to avoid redundant computations.

B.3 DATA STATEMENT

Following Bender and Friedman (2018), we provide a data statement, which documents the generation and provenance of labelled and unlabelled documents in the Reddit Time Corpus (RTC).

A. CURATION RATIONALE The purpose of RTC is to enable our analysis of temporal adaptation of pre-trained language models and downstream task performance. RTC comprises text comments that were posted to the social media site Reddit between March 2017 and February 2020. The unlabelled portion of RTC consists of 36 million comments sampled from *r/news* and *r/worldnews*, two of the most active subreddits (i.e. sub-forums) on the site, which

are dedicated to discussion of current news events. The labelled portion of RTC consists of 0.9 million comments sampled in equal proportions from five subreddits for political discussion (*r/the_donald*, *r/libertarian*, *r/conservative*, *r/politics* and *r/chapotraphouse*). Comments are labelled based on which subreddit they are from. All data is split into 36 evenly-sized subsets based on comment timestamps.

B. LANGUAGE VARIETY RTC only contains English-language text documents, as determined by the `langdetect` Python library. We opted for English language due to data availability. Further, all data in RTC is sourced from Reddit. We consider this a limitation of our analysis and suggest expansion to other languages and data sources as a priority for future research.

C. SPEAKER DEMOGRAPHICS The speakers in RTC are a sample of all Reddit users who posted a comment to one of the seven subreddits covered by RTC between March 2017 and February 2020. In February 2020, *r/worldnews* had around 23.1m subscribers, *r/news* 19.9m, *r/politics* 5.76m, *r/the_donald* 0.79m, *r/libertarian* 0.36m, *r/conservative* 0.30m and *r/chapotraphouse* 0.15m. Reddit does not make information on user demographics available but a February 2019 survey of US users indicated that roughly two-thirds were male, and that user age was skewed towards 18 to 29 years (Statista, 2019).

D. ANNOTATOR DEMOGRAPHICS We did not employ any annotators. All labels in RTC are based on comment metadata, specifically which subreddit a given comment is from.

E. SPEECH SITUATION All comments in RTC were posted to Reddit between March 1st 2017 and February 29th 2020. The intended audience is other subreddit users and site visitors.

F. TEXT CHARACTERISTICS All documents are individual text comments. Pre-processing steps are described in §4.3.2. For the labelled portion of RTC, we provide a label based on which of the five political subreddits in RTC they were posted to. The class distribution is balanced in RTC overall and in each monthly subset.

B.4 Model Training & Parameters

Model Architecture We implemented uncased BERT-base models (Devlin et al., 2019) using the `transformers` Python library (Wolf et al., 2020). Uncased BERT-base, which is trained on lower-cased English text, has 12 layers, a hidden layer size of 768, 12 attention heads and a total of 110 million parameters. For PSP, we added a linear layer with softmax output.

Training Parameters For both MLM and PSP, we used cross-entropy loss. As an optimiser, we used AdamW (Loshchilov and Hutter, 2019) with a 5e-5 learning rate and a 0.01 weight decay. For regularisation, we set a 10% dropout probability. Maximum input sequence length is 128 tokens. For adapting to unlabelled data, we trained for one epoch, i.e. one pass over all additional data, which matches Gururangan et al. (2020). Training batch size was 128. For fine-tuning on labelled data, we trained for three epochs with a batch size of 32, which corresponds to default settings recommended by Devlin et al. (2019). For comparability, we used these same untuned hyperparameters across all experiments.

Computation All experiments were run between March and May 2021 using Nvidia Tesla K80 and V100 GPUs accessed through the University of Oxford's Advanced Research Computing service. Runtime varied from experiment to experiment. Adapting BERT to one million comments for one epoch took around three hours on a V100. Fine-tuning BERT on 20,000 comments for three epochs took around 15 minutes.

Source Code We make all our code available at https://github.com/paul-rottger/temporal-adaptation.

C Appendices for Chapter 5: Annotation Paradigms

C.1 Overview of Subjective Task Datasets

This appendix gives an illustrative overview of how existing NLP dataset work for hate speech and other forms of abuse has (or has not) engaged with annotator subjectivity. For reasons of scope, we focus on 10 English-language datasets. Entries are sorted from most descriptive to most prescriptive annotation.

Sap et al. (2019) and Sap et al. (2022) annotate toxicity. They do not state explicitly that they encourage annotator subjectivity, but their annotation prompts clearly do. They use up to 641 annotators per entry. Overall, they are **very aligned with the descriptive paradigm**.

Cercas Curry et al. (2021) annotate abuse. They gather 'views of expert annotators' based on guidelines that allow for significant subjectivity and do not attempt to resolve disagreements, but also do not explicitly encourage annotator subjectivity. On average, they use around three annotators per entry. Overall, they are **moderately aligned with the descriptive paradigm**.

Talat and Hovy (2016) annotate hate speech. They provide annotators with 11 fine-grained criteria for hate speech, but several criteria invite subjective responses (e.g., 'uses a *problematic* hashtag'). They use up to three annotators per entry. Overall, they are **not clearly aligned with either paradigm**.

Davidson et al. (2017) annotate hate speech. They provide annotators with a brief definition of hate speech and an explanatory paragraph, but their definition also includes subjective criteria like 'intent'. They use three annotators per entry for most of their data. Overall, they are **not clearly aligned with either paradigm**.

Zampieri et al. (2019a) annotate offensive content. They provide annotators with some formal criteria for offensiveness (e.g., 'use of profanity'), but as a whole their guidelines are very brief. They use up to three annotators per entry. Overall, they are **moderately aligned with the**

prescriptive paradigm.

Founta et al. (2018) annotate abuse. They provide annotators with fine-grained definitions for each category and iterate on their taxonomy to facilitate more agreement, but do not share comprehensive guidelines. They use five annotators per entry. Overall, they are **moderately aligned with the prescriptive paradigm**.

Caselli et al. (2020) annotate abuse. They provide annotators with a brief fine-grained decision tree with the explicit intent of reducing annotator subjectivity, and discuss disagreements to resolve them. They use up to three annotators per entry. Overall, they are **moderately aligned with the prescriptive paradigm**.

Vidgen et al. (2021c) annotate hate speech. They provide annotators with fine-grained definitions for each category as well as very detailed annotation guidelines, and disagreements are resolved by an expert. They use up to three annotators per entry. Overall, they are **very aligned with the prescriptive paradigm**.

Vidgen et al. (2021b) annotate abuse. They provide annotators with fine-grained definitions for each category as well as very detailed annotation guidelines, and they use expert-driven group adjudication to resolve disagreements. They use up to three annotators per entry. Overall, they are **very aligned with the prescriptive paradigm**.

C.2 Data Statement

Following Bender and Friedman (2018), we provide a data statement, which documents the generation and provenance of the dataset used for our annotation experiment.

A. CURATION RATIONALE To create our dataset, we sampled 200 Twitter posts from a larger corpus annotated for hateful content by Davidson et al. (2017). Of the posts we sampled, 100 were originally annotated as hateful and 100 as non-hateful. Davidson et al. (2017) had three annotators label most of their data. We sampled only from those posts that had some disagreement among their annotators (i.e., two out of three rather than unanimous agreement),

to encourage disagreement in our experiment. The purpose of our 200-post dataset is to enable the annotation experiment presented in Section 5.4, which illustrates the contrast between the descriptive and prescriptive data annotation paradigms.

B. LANGUAGE VARIETY The dataset contains English-language text posts only.

C. SPEAKER DEMOGRAPHICS All speakers are Twitter users. Davidson et al. (2017) do not share any information on their demographics.

D. ANNOTATOR RECRUITMENT We recruited three groups of 20 annotators using Amazon's Mechanical Turk crowdsourcing marketplace.[29] Annotators were made aware that the task contained instances of offensive language before starting their work, and they could withdraw at any point throughout the work.

E. ANNOTATOR DEMOGRAPHICS All annotators were at least 18 years old when they started their work, and we recruited only annotators that were based in the UK. This was to facilitate comparability across groups of annotators. For each group, we recruited 10 male and 10 female annotators, based on self-reported gender. This was to encourage disagreement within groups, based on the assumption that men would on average disagree more about hateful content with women than with other men, and vice versa. No further annotator demographics were recorded.

F. ANNOTATOR COMPENSATION All annotators were compensated for their work at a rate of at least £16 per hour. The rate was set 50% above the local living wage (£10.85), although all work was completed remotely.

G. SPEECH SITUATION All entries in our dataset were originally posted to the social media site Twitter and then collected by Davidson et al. (2017), who do not share when the posts were made.

[29] https://www.mturk.com/

H. TEXT CHARACTERISTICS All entries in our dataset are individual Twitter text posts, with a length of 140 characters or less. We perform only minimal text cleaning, replacing user mentions (e.g., "@Obama") with "[USER]" and URLs with "[URL]".

I. LICENSE Davidson et al. (2017) make the Twitter data they collected available for further research use via GitHub under an MIT license.[30] Our re-annotated subset of the data is made available under CC-by 4.0 license at https://github.com/paul-rottger/annotation-paradigms, so that the results of our experiment can be reproduced.

J. ETHICS APPROVAL We received approval for our experiment and the data annotation it entailed from our institution's ethics review board.

C.3 ANNOTATION PROMPTS

The three groups of annotators in our experiment all annotated the same data in the same order, but each group received different annotation prompts. The full annotation guidelines for **G2** are available at https://github.com/paul-rottger/annotation-paradigms.

G1 - DESCRIPTIVE GROUP "Imagine you come across the post below on social media. **Do you personally feel this post is hateful?** We want to understand your own opinions, so try to disregard any impressions you might have about whether other people would find it hateful."

G2 - PRESCRIPTIVE GROUP "Imagine you come across the post below on social media. **Does this post meet the criteria for hate speech?** We are trying to collect objective judgments, so try to disregard any feelings you might have about whether you personally find it hateful.

Click here to view the criteria: LINK"

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

G3 - CONTROL GROUP "Imagine you come across the post below on social media. **Does this post meet the criteria for hate speech?** A post is considered hate speech if it is 1) abusive and 2) targeted against a protected group (e.g., women) or at its members for being a part of that group."

D Appendices for Chapter 6: Data-Efficient Low-Resource Hate Detection

D.1 Details on Model Training

Parameters We implemented XLM-T as well as the five language-specific monolingual models (§6.2.2) using the HuggingFace `transformers` library (Wolf et al., 2020). Training batch size was 16. Maximum sequence length was 128 tokens. Otherwise, we used default parameters. The optional first phase of fine-tuning on English data was for three epochs. Fine-tuning on the target language was for five epochs, with subsequent best epoch selection based on macro F1 on the held-out development set.

Computation We ran all experiments on 16-core cloud CPUs on the University of Oxford's Advanced Research Computing cluster. Fine-tuning 120 models for one of the five target languages took around four hours for monolingual models and around seven hours for XLM-T.

D.2 Macro F1 on Held-Out Test Sets

For larger versions of the graphs in Figure 6 from the main body, see Figure 8 below. See Table 21 for a version of Table 14 across all N.

D.3 Macro F1 on MHC

For larger versions of the graphs in Figure 7 from the main body, see Figure 9 below. Similarly, see Table 22 for a version of Table 15 across all N.

D.4 Regression Results for All Models

See Table 23 below.

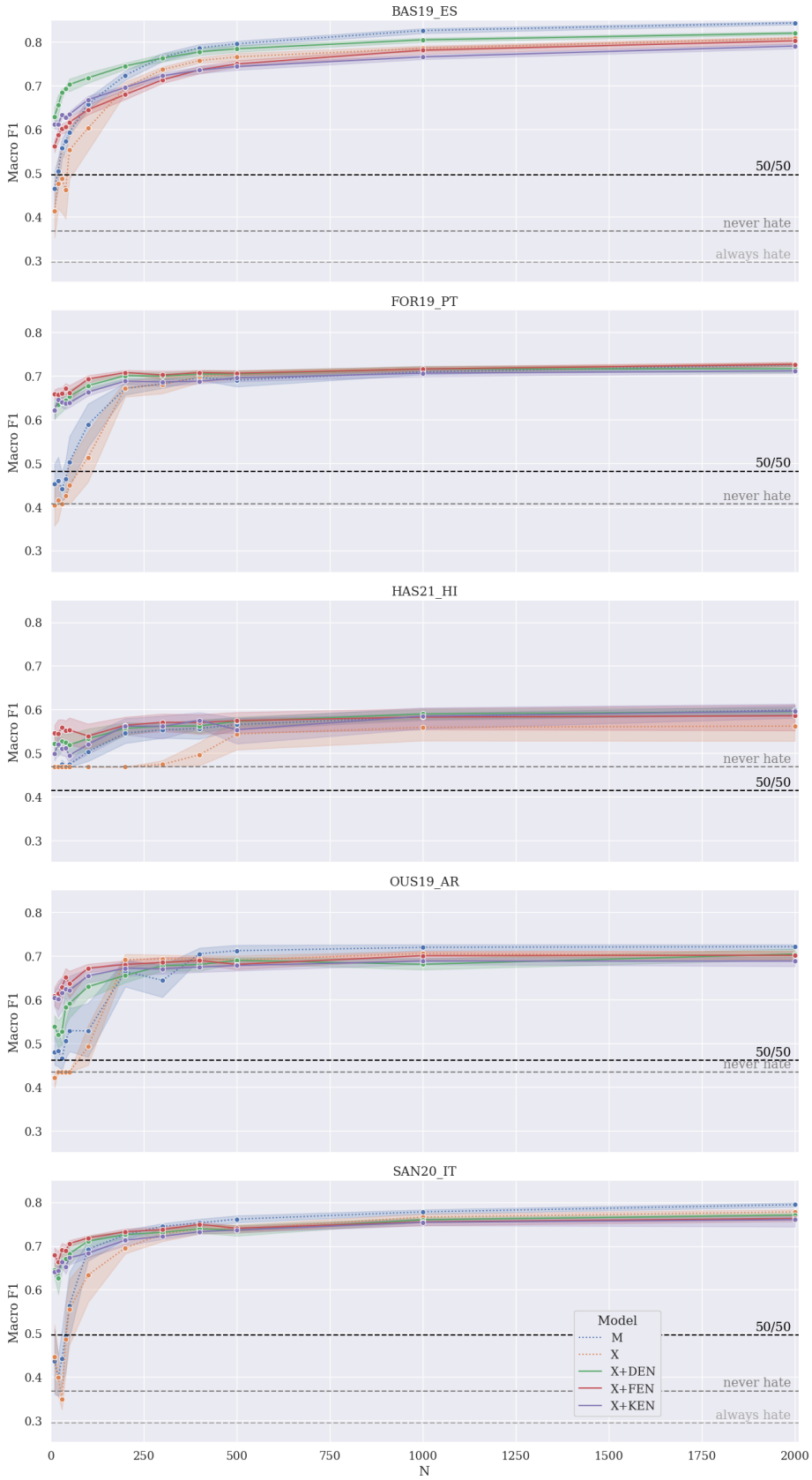

Figure 8: Macro F1 on target-language held-out test sets across models trained on up to N=2,000 target-language entries. Model notation as described in §6.2.2. Confidence intervals and model baselines as described in §6.2.4.

BAS19_ES

N	10	20	30	40	50	100	200	300	400	500	1000	2,000
M	0.46	0.50	0.56	0.57	0.59	0.66	0.72	**0.77**	**0.79**	**0.80**	**0.83**	**0.84**
X	0.41	0.48	0.49	0.46	0.55	0.60	0.70	0.74	0.76	0.77	0.79	0.81
X+DEN	**0.63**	**0.66**	**0.68**	**0.69**	**0.70**	**0.72**	**0.75**	0.76	0.78	0.78	0.81	0.82
X+FEN	0.56	0.59	0.60	0.61	0.62	0.65	0.68	0.71	0.74	0.75	0.78	0.80
X+KEN	0.61	0.61	0.63	0.63	0.63	0.67	0.70	0.72	0.74	0.74	0.77	0.79

FOR19_PT

N	10	20	30	40	50	100	200	300	400	500	1000	2,000
M	0.45	0.46	0.44	0.46	0.50	0.59	0.67	0.68	0.70	0.69	0.71	**0.73**
X	0.40	0.42	0.41	0.42	0.45	0.51	0.67	0.68	0.70	0.70	0.71	**0.73**
X+DEN	0.62	0.63	0.64	0.64	0.65	0.68	0.70	**0.70**	0.70	0.70	**0.72**	0.72
X+FEN	**0.66**	**0.66**	**0.66**	**0.67**	**0.66**	**0.69**	**0.71**	**0.70**	**0.71**	**0.71**	**0.72**	**0.73**
X+KEN	0.62	0.65	0.64	0.64	0.64	0.66	0.69	0.69	0.69	0.70	0.71	0.71

HAS21_HI

N	10	20	30	40	50	100	200	300	400	500	1000	2,000
M	0.47	0.47	0.47	0.47	0.47	0.50	0.55	0.55	0.56	**0.57**	0.58	**0.60**
X	0.47	0.47	0.47	0.47	0.47	0.47	0.47	0.47	0.50	0.54	0.56	0.56
X+DEN	0.52	0.52	0.53	0.53	0.52	0.53	**0.56**	0.56	0.56	**0.57**	**0.59**	**0.60**
X+FEN	**0.55**	**0.55**	**0.56**	**0.55**	**0.55**	**0.54**	**0.56**	**0.57**	**0.57**	**0.57**	0.58	0.59
X+KEN	0.50	0.52	0.51	0.51	0.49	0.52	**0.56**	0.56	0.58	0.55	0.58	**0.60**

OUS19_AR

N	10	20	30	40	50	100	200	300	400	500	1000	2,000
M	0.48	0.48	0.46	0.51	0.53	0.53	0.66	0.64	**0.70**	**0.71**	**0.72**	**0.72**
X	0.42	0.43	0.43	0.43	0.43	0.49	**0.69**	**0.69**	0.69	0.69	0.71	0.70
X+DEN	0.54	0.52	0.53	0.58	0.59	0.63	0.66	0.68	0.68	0.69	0.68	0.70
X+FEN	**0.61**	**0.61**	**0.63**	**0.65**	**0.64**	**0.67**	0.68	**0.69**	0.69	0.68	0.70	0.70
X+KEN	0.60	0.60	0.62	0.62	0.62	0.65	0.67	0.67	0.67	0.68	0.69	0.69

SAN20_IT

N	10	20	30	40	50	100	200	300	400	500	1000	2,000
M	0.44	0.40	0.44	0.50	0.56	0.69	**0.73**	**0.75**	**0.75**	**0.76**	**0.78**	**0.79**
X	0.45	0.40	0.35	0.49	0.56	0.63	0.70	0.73	0.74	0.74	0.77	0.78
X+DEN	0.64	0.63	0.67	0.67	0.68	0.71	**0.73**	0.73	0.74	0.74	0.76	0.77
X+FEN	**0.68**	**0.66**	**0.69**	**0.69**	**0.71**	**0.72**	**0.73**	0.74	**0.75**	0.74	0.76	0.76
X+KEN	0.64	0.64	0.66	0.65	0.67	0.68	0.71	0.72	0.73	0.74	0.75	0.76

Table 21: Macro F1 on respective held-out test sets for models fine-tuned on N target-language entries, averaged across 10 random seeds for each N. Best performance for a given N in **bold**.

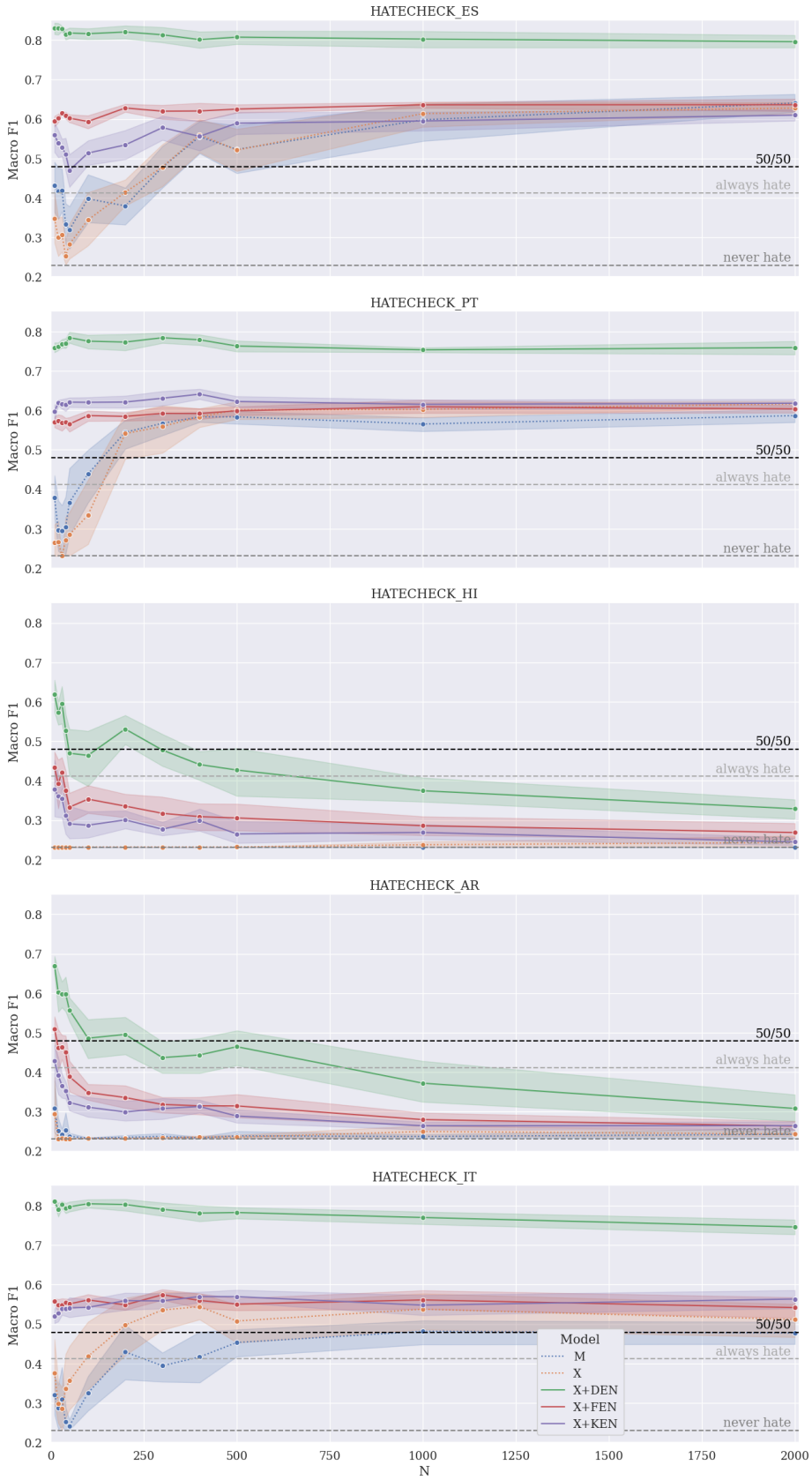

Figure 9: Macro F1 on target-language tests from MHC across models trained on up to N = 2,000 target-language entries. Model notation as described in §6.2.2. Confidence intervals and model baselines as described in §6.2.4.

HATECHECK_ES

N	10	20	30	40	50	100	200	300	400	500	1000	2,000
M	0.43	0.42	0.42	0.33	0.32	0.40	0.38	0.48	0.56	0.52	0.60	0.64
X	0.35	0.30	0.31	0.25	0.28	0.34	0.41	0.48	0.56	0.52	0.61	0.63
X+DEN	**0.83**	**0.83**	**0.83**	**0.82**	**0.82**	**0.82**	**0.82**	**0.81**	**0.80**	**0.81**	**0.80**	**0.80**
X+FEN	0.59	0.60	0.61	0.61	0.60	0.59	0.63	0.62	0.62	0.63	0.64	0.64
X+KEN	0.56	0.54	0.53	0.51	0.47	0.51	0.53	0.58	0.56	0.59	0.59	0.61

HATECHECK_PT

N	10	20	30	40	50	100	200	300	400	500	1000	2,000
M	0.38	0.30	0.30	0.31	0.37	0.44	0.54	0.57	0.58	0.58	0.57	0.59
X	0.27	0.27	0.23	0.27	0.28	0.33	0.54	0.56	0.58	0.60	0.60	0.62
X+DEN	**0.76**	**0.76**	**0.77**	**0.77**	**0.78**	**0.78**	**0.77**	**0.78**	**0.78**	**0.76**	**0.75**	**0.76**
X+FEN	0.57	0.57	0.57	0.57	0.57	0.59	0.58	0.59	0.59	0.60	0.61	0.60
X+KEN	0.60	0.62	0.62	0.62	0.62	0.62	0.62	0.63	0.64	0.62	0.62	0.62

HATECHECK_HI

N	10	20	30	40	50	100	200	300	400	500	1000	2,000
M	0.23	0.23	0.23	0.23	0.23	0.23	0.23	0.23	0.23	0.23	0.23	0.23
X	0.23	0.23	0.23	0.23	0.23	0.23	0.23	0.23	0.23	0.23	0.24	0.24
X+DEN	**0.62**	**0.57**	**0.60**	**0.53**	**0.47**	**0.46**	**0.53**	**0.48**	**0.44**	**0.43**	**0.37**	**0.33**
X+FEN	0.43	0.39	0.42	0.37	0.33	0.35	0.34	0.32	0.31	0.31	0.29	0.27
X+KEN	0.38	0.36	0.35	0.31	0.29	0.29	0.30	0.28	0.30	0.27	0.27	0.24

HATECHECK_AR

N	10	20	30	40	50	100	200	300	400	500	1000	2,000
M	0.31	0.25	0.24	0.25	0.24	0.23	0.23	0.24	0.23	0.24	0.24	0.24
X	0.29	0.23	0.23	0.23	0.23	0.23	0.23	0.23	0.24	0.24	0.25	0.24
X+DEN	**0.67**	**0.60**	**0.60**	**0.60**	**0.56**	**0.49**	**0.50**	**0.44**	**0.44**	**0.46**	**0.37**	**0.31**
X+FEN	0.51	0.46	0.46	0.45	0.39	0.35	0.34	0.32	0.31	0.31	0.28	0.26
X+KEN	0.43	0.39	0.37	0.35	0.32	0.31	0.30	0.31	0.31	0.29	0.26	0.26

HATECHECK_IT

N	10	20	30	40	50	100	200	300	400	500	1000	2,000
M	0.32	0.29	0.31	0.25	0.24	0.33	0.43	0.39	0.42	0.45	0.48	0.48
X	0.38	0.30	0.29	0.34	0.36	0.42	0.50	0.54	0.54	0.51	0.54	0.51
X+DEN	**0.81**	**0.79**	**0.80**	**0.79**	**0.80**	**0.80**	**0.80**	**0.79**	**0.78**	**0.78**	**0.77**	**0.75**
X+FEN	0.56	0.55	0.55	0.55	0.55	0.56	0.55	0.57	0.56	0.55	0.56	0.54
X+KEN	0.52	0.53	0.54	0.54	0.54	0.54	0.56	0.56	0.57	0.57	0.55	0.56

Table 22: Macro F1 on target-language HateCheck for models fine-tuned on N target-language entries, averaged across 10 random seeds for each N. Best performance for a given N in **bold**.

Model: M

Test Set	β_0	β_1	R^2	$2N$
BAS19_ES	0.2879	0.0792	91.97	5.49
FOR19_PT	0.2628	0.0670	71.23	4.65
HAS21_HI	0.2998	0.0607	69.11	4.21
OUS19_ES	0.2092	0.0866	70.08	6.00
SAN20_IT	0.3784	0.0294	77.68	2.04

Model: X

Test Set	β_0	β_1	R^2	$2N$
BAS19_ES	0.2107	0.0856	75.39	5.93
FOR19_PT	0.1727	0.0808	79.17	5.6
HAS21_HI	0.1727	0.0808	79.17	5.6
OUS19_ES	0.1852	0.0869	69.09	6.02
SAN20_IT	0.4004	0.019	40.73	1.32

Model: X+DEN

Test Set	β_0	β_1	R^2	$2N$
BAS19_ES	0.5543	0.0363	87.75	2.52
FOR19_PT	0.5756	0.0205	63.66	1.42
HAS21_HI	0.4691	0.0163	43.91	1.13
OUS19_ES	0.4320	0.0391	64.99	2.71
SAN20_IT	0.5724	0.0272	69.42	1.89

Model: X+FEN

Test Set	β_0	β_1	R^2	$2N$
BAS19_ES	0.4347	0.0489	93.95	3.39
FOR19_PT	0.6167	0.0148	60.96	1.02
HAS21_HI	0.5684	0.0192	45.73	1.33
OUS19_ES	0.6256	0.0191	63.71	1.32
SAN20_IT	0.5227	0.008	7.81	0.56

Model: X+KEN

Test Set	β_0	β_1	R^2	$2N$
BAS19_ES	0.5011	0.0381	93.59	2.64
FOR19_PT	0.5797	0.0182	72.81	1.26
HAS21_HI	0.5545	0.0195	40.66	1.35
OUS19_ES	0.5712	0.026	82.84	1.8
SAN20_IT	0.4444	0.0198	42.5	1.37

Table 23: OLS results for all models in our experiments, expanding on Table 16 from the main body. All regression coefficients β_0 and β_1 are significant at $p < 0.001$.

E Co-Authorship and Contributions

My thesis comprises edited versions of four published, co-authored research articles. I was first author on all four articles and, as stipulated by University of Oxford guidelines, my work contributed the majority to each article. My contributions on each article are outlined in more detail below. My co-authors all signed off on this breakdown ahead of thesis submission.

"HateCheck - Functional Tests for Hate Speech Detection Models" was authored by **Paul Röttger**, Bertie Vidgen, Dong Nguyen, Zeerak Talat, Helen Margetts, and Janet B. Pierrehumbert. It was published at **ACL 2021** (Main Conference). The citation is Röttger et al. (2021). All code is available at github.com/paul-rottger/hatecheck-experiments and github.com/paul-rottger/hatecheck-data. I developed the initial idea, after which Bertie Vidgen and I initiated the project. We conducted all interviews. Janet B. Pierrehumbert provided guidance on theoretical framing and the interpretability of language systems. I created the test suite and conducted all experiments and analyses. I wrote the paper, with feedback on structure and references from Janet B. Pierrehumbert. The other authors gave feedback at major milestones (test suite creation, arXiv submission, conference submission).

"Temporal Adaptation of BERT and Performance on Downstream Document Classification: Insights from Social Media" was authored by **Paul Röttger** and Janet B. Pierrehumbert. It was published at **EMNLP 2021** (Findings). The citation is Röttger and Pierrehumbert (2021). All code is available at https://github.com/paul-rottger/temporal-adaptation. I developed the initial idea, after which Janet B. Pierrehumbert and I initiated the project. Janet B. Pierrehumbert provided key references on language change from computational linguistics. I devised the experimental setup, and conducted all experiments and analyses, discussing results with Janet B. Pierrehumbert. I wrote the paper, with comments from Janet B. Pierrehumbert along the way.

"Two Contrasting Data Annotation Paradigms for Subjective NLP Tasks" was authored by **Paul Röttger**, Bertie Vidgen, Dirk Hovy, and Janet B. Pierrehumbert. It was published at **NAACL 2022** (Main Conference). The citation is Röttger et al. (2022c). All code is available at github.com/paul-rottger/annotation-paradigms. I developed the initial idea and initiated the

project. I devised and conducted the annotation experiment, with feedback on experimental design and references from Janet B. Pierrehumbert. I wrote the paper, and all other authors provided regular feedback throughout.

"Data-Efficient Strategies for Expanding Hate Speech Detection into Under-Resourced Languages" was authored by **Paul Röttger**, Debora Nozza, Federico Bianchi, and Dirk Hovy. It was published at **EMNLP 2022** (Main Conference). The citation is Röttger et al. (2022a). All code is available at github.com/paul-rottger/efficient-low-resource-hate-detection. I developed the initial idea, after which Debora Nozza, Federico Bianchi, Dirk Hovy and I initiated the project. I devised the experimental setup, sought feedback from the other authors, and conducted all experiments and analyses. I wrote the paper, with comments and suggestions from the other authors along the way.

F Funding Statement

Personal Funding

My DPhil at the University of Oxford was primarily funded by the German Academic Scholarship Foundation, from which I received a full scholarship covering my tuition and living costs. I also received a smaller amount of funding from the Alan Turing Institute's Enrichment Scheme, which supported my research visit there from October 2021 to March 2022. The Alan Turing Institute also covered my travel and attendance at NAACL 2022 in Seattle. Bocconi University covered my travel to and from my 2022 research visit to Dirk Hovy's group in Milan.

Co-Authors

HateCheck Bertram Vidgen and Helen Margetts were supported by Wave 1 of the UKRI Strategic Priorities Fund under the EPSRC Grant EP/T001569/1, particularly the "Criminal Justice System" theme within that grant, and the "Hate Speech: Measures & Counter-Measures" project at The Alan Turing Institute. Dong Nguyen was supported by the "Digital Society - The Informed Citizen" research programme, which is (partly) financed by the Dutch Research Council (NWO), project 410.19.007. Zeerak Talat was supported in part by the Canada 150 Research Chair program and the UK-Canada AI Artificial Intelligence Initiative. Janet B. Pierrehumbert was supported by EPSRC Grant EP/T023333/1.

Temp. Adaptation Janet B. Pierrehumbert was supported by EPSRC Grant EP/T023333/1.

Annotation Paradigms Bertie Vidgen was supported by The Alan Turing Institute and Towards Turing 2.0 under the EPSRC Grant EP/W037211/1. Dirk Hovy received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (grant agreement No. 949944). Janet B. Pierrehumbert was supported by EPSRC Grant EP/T023333/1.

DATA-EFFICIENT LOW-RESOURCE HATE DETECTION Debora Nozza received funding from Fondazione Cariplo (grant No. 2020-4288, MONICA). Dirk Hovy received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (grant agreement No. 949944).

REFERENCES

Oshin Agarwal and Ani Nenkova. 2022. Temporal effects on pre-trained models for language processing tasks. *Transactions of the Association for Computational Linguistics*, 10:904–921.

Hwijeen Ahn, Jimin Sun, Chan Young Park, and Jungyun Seo. 2020. NLPDove at SemEval-2020 task 12: Improving offensive language detection with cross-lingual transfer. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 1576–1586, Barcelona (online). International Committee for Computational Linguistics.

Sohail Akhtar, Valerio Basile, and Viviana Patti. 2020. Modeling annotator perspective and polarized opinions to improve hate speech detection. In *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, volume 8, pages 151–154.

Sohail Akhtar, Valerio Basile, and Viviana Patti. 2021. Whose opinions matter? perspective-aware models to identify opinions of hate speech victims in abusive language detection. *arXiv preprint arXiv:2106.15896*.

Hala Al Kuwatly, Maximilian Wich, and Georg Groh. 2020. Identifying and measuring annotator bias based on annotators' demographic characteristics. In *Proceedings of the Fourth Workshop on Online Abuse and Harms*, pages 184–190, Online. Association for Computational Linguistics.

Abeer AlDayel and Walid Magdy. 2021. Stance detection on social media: State of the art and trends. *Information Processing & Management*, 58(4):102597.

Cecilia Ovesdotter Alm. 2011. Subjective natural language problems: Motivations, applications, characterizations, and implications. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 107–112, Portland, Oregon, USA. Association for Computational Linguistics.

Emily Alsentzer, John Murphy, William Boag, Wei-Hung Weng, Di Jindi, Tristan Naumann, and Matthew McDermott. 2019. Publicly available clinical BERT embeddings. In *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, pages 72–78, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Eduardo G Altmann, Janet B Pierrehumbert, and Adilson E Motter. 2009. Beyond word frequency: Bursts, lulls, and scaling in the temporal distributions of words. *PLOS one*, 4(11):e7678.

Sai Saketh Aluru, Binny Mathew, Punyajoy Saha, and Animesh Mukherjee. 2020. A deep dive into multilingual hate speech classification. In *Machine Learning and Knowledge Discovery in Databases. Applied Data Science and Demo Track: European Conference, ECML PKDD 2020, Ghent, Belgium, September 14–18, 2020, Proceedings, Part V*, page 423–439, Berlin, Heidelberg. Springer-Verlag.

Wissam Antoun, Fady Baly, and Hazem Hajj. 2020. AraBERT: Transformer-based model for Arabic language understanding. In *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection*, pages 9–15, Marseille, France. European Language Resource Association.

Jordi Armengol-Estapé, Casimiro Pio Carrino, Carlos Rodriguez-Penagos, Ona de Gibert Bonet, Carme Armentano-Oller, Aitor Gonzalez-Agirre, Maite Melero, and Marta Villegas. 2021. Are multilingual models the best choice for moderately under-resourced languages? A comprehensive assessment for Catalan. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4933–4946, Online. Association for Computational Linguistics.

Ishaan Arora, Julia Guo, Sarah Ita Levitan, Susan McGregor, and Julia Hirschberg. 2020. A novel methodology for developing automatic harassment classifiers for Twitter. In *Proceedings of the Fourth Workshop on Online Abuse and Harms*, pages 7–15, Online. Association for Computational Linguistics.

Lora Aroyo and Chris Welty. 2015. Truth is a lie: Crowd truth and the seven myths of human annotation. *AI Magazine*, 36(1):15–24.

Mikel Artetxe and Holger Schwenk. 2019. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *Transactions of the Association for Computational Linguistics*, 7:597–610.

Giuseppe Attanasio, Debora Nozza, Dirk Hovy, and Elena Baralis. 2022. Entropy-based attention regularization frees unintended bias mitigation from lists. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1105–1119, Dublin, Ireland. Association for Computational Linguistics.

Pinkesh Badjatiya, Shashank Gupta, Manish Gupta, and Vasudeva Varma. 2017. Deep learning for hate speech detection in tweets. In *Proceedings of the 26th International Conference on World Wide Web Companion*, pages 759–760.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, et al. 2022. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*.

Somnath Banerjee, Maulindu Sarkar, Nancy Agrawal, Punyajoy Saha, and Mithun Das. 2021. Exploring transformer based models to identify hate speech and offensive content in English and Indo-Aryan languages. *arXiv preprint arXiv:2111.13974*.

Michele Banko, Brendon MacKeen, and Laurie Ray. 2020. A Unified Taxonomy of Harmful Content. In *Proceedings of the Fourth Workshop on Online Abuse and Harms*, pages 125–137. Association for Computational Linguistics.

Francesco Barbieri, Luis Espinosa Anke, and Jose Camacho-Collados. 2022. XLM-T: Multilingual language models in Twitter for sentiment analysis and beyond. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 258–266, Marseille, France. European Language Resources Association.

Marion Bartl, Malvina Nissim, and Albert Gatt. 2020. Unmasking contextual stereotypes: Measuring and mitigating BERT's gender bias. In *Proceedings of the Second Workshop on Gender Bias in Natural Language Processing*, pages 1–16, Barcelona, Spain (Online). Association for Computational Linguistics.

Valerio Basile, Cristina Bosco, Elisabetta Fersini, Debora Nozza, Viviana Patti, Francisco Manuel Rangel Pardo, Paolo Rosso, and Manuela Sanguinetti. 2019. SemEval-2019 task 5: Multilingual detection of hate speech against immigrants and women in Twitter. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 54–63, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Valerio Basile, Federico Cabitza, Andrea Campagner, and Michael Fell. 2021a. Toward a perspectivist turn in ground truthing for predictive computing. *Conference of the Italian Chapter of the Association for Intelligent Systems (ItAIS 2021)*.

Valerio Basile, Michael Fell, Tommaso Fornaciari, Dirk Hovy, Silviu Paun, Barbara Plank, Massimo Poesio, and Alexandra Uma. 2021b. We need to consider disagreement in evaluation. In *Proceedings of the 1st Workshop on Benchmarking: Past, Present and Future*, pages 15–21, Online. Association for Computational Linguistics.

Jason Baumgartner, Savvas Zannettou, Brian Keegan, Megan Squire, and Jeremy Blackburn. 2020. The Pushshift Reddit dataset. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 14, pages 830–839.

BBC. 2020. Facebook to pay $52m to content moderators over ptsd. *BBC News*.

Boris Beizer. 1995. *Black-box testing: techniques for functional testing of software and systems*. John Wiley & Sons, Inc.

Yonatan Belinkov and Yonatan Bisk. 2018. Synthetic and natural noise both break neural machine translation. In *Proceedings of the 6th International Conference on Learning Representations*.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A pretrained language model for scientific text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3615–3620, Hong Kong, China. Association for Computational Linguistics.

Emily M. Bender and Batya Friedman. 2018. Data statements for natural language processing: Toward mitigating system bias and enabling better science. *Transactions of the Association for Computational Linguistics*, 6:587–604.

Emily M. Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '21, page 610–623, New York, NY, USA. Association for Computing Machinery.

Irina Bigoulaeva, Viktor Hangya, and Alexander Fraser. 2021. Cross-lingual transfer learning for hate speech detection. In *Proceedings of the First Workshop on Language Technology for Equality, Diversity and Inclusion*, pages 15–25, Kyiv. Association for Computational Linguistics.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022.

Pete Burnap and Matthew L Williams. 2015. Cyber hate speech on Twitter: An application of machine classification and statistical modeling for policy and decision making. *Policy & Internet*, 7(2):223–242.

Peter Burnap and Matthew Leighton Williams. 2014. Hate speech, machine classification and statistical modelling of information flows on twitter: Interpretation and communication for policy decision making. *Proceedings of the 2014 Internet, Policy and Politics Conference*.

Agostina Calabrese, Michele Bevilacqua, Björn Ross, Rocco Tripodi, and Roberto Navigli. 2021. Aaa: Fair evaluation for abuse detection systems wanted. In *13th ACM Web Science Conference 2021*, pages 243–252.

Rui Cao, Roy Ka-Wei Lee, and Tuan-Anh Hoang. 2020. DeepHate: Hate speech detection via multi-faceted text representations. In *Proceedings of the 12th ACM Conference on Web Science*, pages 11–20.

Dallas Card, Amber E. Boydstun, Justin H. Gross, Philip Resnik, and Noah A. Smith. 2015. The media frames corpus: Annotations of frames across issues. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 438–444, Beijing, China. Association for Computational Linguistics.

Tommaso Caselli, Valerio Basile, Jelena Mitrović, and Michael Granitzer. 2021. HateBERT: Retraining BERT for abusive language detection in English. In *Proceedings of the 5th Workshop on Online Abuse and Harms (WOAH 2021)*, pages 17–25, Online. Association for Computational Linguistics.

Tommaso Caselli, Valerio Basile, Jelena Mitrović, Inga Kartoziya, and Michael Granitzer. 2020. I feel offended, don't be abusive! implicit/explicit messages in offensive and abusive language. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 6193–6202, Marseille, France. European Language Resources Association.

Amanda Cercas Curry, Gavin Abercrombie, and Verena Rieser. 2021. ConvAbuse: Data, analysis, and benchmarks for nuanced detection in conversational AI. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7388–7403, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Tuhin Chakrabarty, Christopher Hidey, and Kathy McKeown. 2019. IMHO fine-tuning improves claim detection. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 558–563, Minneapolis, Minnesota. Association for Computational Linguistics.

Yung-Sung Chuang, Mingye Gao, Hongyin Luo, James Glass, Hung-yi Lee, Yun-Nung Chen, and Shang-Wen Li. 2021. Mitigating biases in toxic language detection through invariant rationalization. In *Proceedings of the 5th Workshop on Online Abuse and Harms (WOAH 2021)*, pages 114–120, Online. Association for Computational Linguistics.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*.

Kenneth W. Church. 2000. Empirical estimates of adaptation: The chance of two Noriegas is closer to $p/2$ than $p2$. In *COLING 2000 Volume 1: The 18th International Conference on Computational Linguistics*.

Kenneth Ward Church and William A Gale. 1995. Poisson mixtures. *Nat. Lang. Eng.*, 1(2):163–190.

Kevin Clark, Minh-Thang Luong, Quoc Le, and Christopher D. Manning. 2020a. Pre-training transformers as energy-based cloze models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 285–294, Online. Association for Computational Linguistics.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020b. ELECTRA: Pre-training text encoders as discriminators rather than generators. In *International Conference on Learning Representations*.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

M. D. Conover, B. Goncalves, J. Ratkiewicz, A. Flammini, and F. Menczer. 2011. Predicting the political alignment of Twitter users. In *2011 IEEE Third International Conference on Privacy, Security, Risk and Trust and 2011 IEEE Third International Conference on Social Computing*, pages 192–199.

Juliet M Corbin and Anselm Strauss. 1990. Grounded theory research: Procedures, canons, and evaluative criteria. *Qualitative Sociology*, 13(1):3–21.

Aida Mostafazadeh Davani, Mohammad Atari, Brendan Kennedy, and Morteza Dehghani. 2021. Hate speech classifiers learn human-like social stereotypes.

Aida Mostafazadeh Davani, Mark Díaz, and Vinodkumar Prabhakaran. 2022. Dealing with disagreements: Looking beyond the majority vote in subjective annotations. *Transactions of the Association for Computational Linguistics*, 10:92–110.

Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. 2017. Automated hate speech detection and the problem of offensive language. In *Proceedings of the 11th International AAAI Conference on Web and Social Media*, pages 512–515. Association for the Advancement of Artificial Intelligence.

Daniel de Vassimon Manela, David Errington, Thomas Fisher, Boris van Breugel, and Pasquale Minervini. 2021. Stereotype and skew: Quantifying gender bias in pre-trained and fine-tuned language models. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2232–2242, Online. Association for Computational Linguistics.

Marco Del Tredici and Raquel Fernández. 2017. Semantic variation in online communities of practice. In *IWCS 2017 - 12th International Conference on Computational Semantics - Long papers*.

Fabio Del Vigna, Andrea Cimino, Felice Dell'Orletta, Marinella Petrocchi, and Maurizio Tesconi. 2017. Hate me, hate me not: Hate speech detection on Facebook. In *Proceedings of the First Italian Conference on Cybersecurity (ITASEC17)*, pages 86–95.

Dorottya Demszky, Nikhil Garg, Rob Voigt, James Zou, Jesse Shapiro, Matthew Gentzkow, and Dan Jurafsky. 2019. Analyzing polarization in social media: Method and application to tweets on 21 mass shootings. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2970–3005, Minneapolis, Minnesota. Association for Computational Linguistics.

Christoph Demus, Jonas Pitz, Mina Schütz, Nadine Probol, Melanie Siegel, and Dirk Labudde. 2022. Detox: A comprehensive dataset for German offensive language and conversation analysis. In *Proceedings of the Sixth Workshop on Online Abuse and Harms (WOAH)*, pages 143–153, Seattle, Washington (Hybrid). Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Bhuwan Dhingra, Jeremy R Cole, Julian Martin Eisenschlos, Daniel Gillick, Jacob Eisenstein, and William W Cohen. 2022. Time-aware language models as temporal knowledge bases. *Transactions of the Association for Computational Linguistics*, 10:257–273.

Mark Diaz, Isaac Johnson, Amanda Lazar, Anne Marie Piper, and Darren Gergle. 2018. *Addressing Age-Related Bias in Sentiment Analysis*, page 1–14. Association for Computing Machinery, New York, NY, USA.

Emily Dinan, Samuel Humeau, Bharath Chintagunta, and Jason Weston. 2019. Build it break it fix it for dialogue safety: Robustness from adversarial human attack. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4537–4546, Hong Kong, China. Association for Computational Linguistics.

Lucas Dixon, John Li, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. 2018. Measuring and mitigating unintended bias in text classification. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pages 67–73. Association for Computing Machinery.

Nemanja Djuric, Jing Zhou, Robin Morris, Mihajlo Grbovic, Vladan Radosavljevic, and Narayan Bhamidipati. 2015. Hate speech detection with comment embeddings. In *24th international WWW conference*, pages 29–30.

Ashwin Geet d'Sa, Irina Illina, and Dominique Fohr. 2020. BERT and fastText embeddings for automatic detection of toxic speech. In *SIIE 2020-Information Systems and Economic Intelligence*.

Vittoria Elliott and Tekendra Parmar. 2020. Report: "the despair and darkness of people will get to you". *Rest of World*.

Allyson Ettinger. 2020. What BERT is not: Lessons from a new suite of psycholinguistic diagnostics for language models. *Transactions of the Association for Computational Linguistics*, 8:34–48.

Komal Florio, Valerio Basile, Marco Polignano, Pierpaolo Basile, and Viviana Patti. 2020. Time of your hate: The challenge of time in hate speech detection on social media. *Applied Sciences*, 10(12):4180.

Tommaso Fornaciari, Alexandra Uma, Silviu Paun, Barbara Plank, Dirk Hovy, and Massimo Poesio. 2021. Beyond black & white: Leveraging annotator disagreement via soft-label multi-task learning. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2591–2597, Online. Association for Computational Linguistics.

Paula Fortuna and Sérgio Nunes. 2018. A survey on automatic detection of hate speech in text. *ACM Computing Surveys (CSUR)*, 51(4):1–30.

Paula Fortuna, João Rocha da Silva, Juan Soler-Company, Leo Wanner, and Sérgio Nunes. 2019. A hierarchically-labeled Portuguese hate speech dataset. In *Proceedings of the Third Workshop on Abusive Language Online*, pages 94–104, Florence, Italy. Association for Computational Linguistics.

Antigoni Founta, Constantinos Djouvas, Despoina Chatzakou, Ilias Leontiadis, Jeremy Blackburn, Gianluca Stringhini, Athena Vakali, Michael Sirivianos, and Nicolas Kourtellis. 2018. Large scale crowdsourcing and characterization of Twitter abusive behavior. *Proceedings of the International AAAI Conference on Web and Social Media*, 12(1).

Foxglove. 2021. Open letter from 60 content moderators to facebook calling for proper mental health support and an end to outsourcing. *Foxglove*.

Björn Gambäck and Utpal Kumar Sikdar. 2017. Using convolutional neural networks to classify hate-speech. In *Proceedings of the first workshop on abusive language online*, pages 85–90.

Matt Gardner, Yoav Artzi, Victoria Basmov, Jonathan Berant, Ben Bogin, Sihao Chen, Pradeep Dasigi, Dheeru Dua, Yanai Elazar, Ananth Gottumukkala, Nitish Gupta, Hannaneh Hajishirzi, Gabriel Ilharco, Daniel Khashabi, Kevin Lin, Jiangming Liu, Nelson F. Liu, Phoebe Mulcaire, Qiang Ning, Sameer Singh, Noah A. Smith, Sanjay Subramanian, Reut Tsarfaty, Eric Wallace, Ally Zhang, and Ben Zhou. 2020. Evaluating models' local decision boundaries via contrast sets. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1307–1323, Online. Association for Computational Linguistics.

Sahaj Garg, Vincent Perot, Nicole Limtiaco, Ankur Taly, Ed H. Chi, and Alex Beutel. 2019. Counterfactual fairness in text classification through robustness. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, AIES '19, page 219–226, New York, NY, USA. Association for Computing Machinery.

Mor Geva, Yoav Goldberg, and Jonathan Berant. 2019. Are we modeling the task or the annotator? an investigation of annotator bias in natural language understanding datasets. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1161–1166, Hong Kong, China. Association for Computational Linguistics.

Tarleton Gillespie. 2018. *Custodians of the Internet: Platforms, content moderation, and the hidden decisions that shape social media*. Yale University Press.

Tarleton Gillespie. 2020. Content moderation, ai, and the question of scale. *Big Data & Society*, 7(2):2053951720943234.

Tarleton Gillespie. 2022. Do not recommend? reduction as a form of content moderation. *Social Media+ Society*, 8(3):20563051221117552.

Njagi Dennis Gitari, Zhang Zuping, Hanyurwimfura Damien, and Jun Long. 2015. A lexicon-based approach for hate speech detection. *International Journal of Multimedia and Ubiquitous Engineering*, 10(4):215–230.

Max Glockner, Vered Shwartz, and Yoav Goldberg. 2018. Breaking NLI systems with sentences that require simple lexical inferences. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 650–655, Melbourne, Australia. Association for Computational Linguistics.

Jennifer Golbeck, Zahra Ashktorab, Rashad O Banjo, Alexandra Berlinger, Siddharth Bhagwan, Cody Buntain, Paul Cheakalos, Alicia A Geller, Rajesh Kumar Gnanasekaran, Raja Rajan Gunasekaran, et al. 2017. A large labeled corpus for online harassment research. In *Proceedings of the 9th ACM Conference on Web Science*, pages 229–233. Association for Computing Machinery.

Mitchell L Gordon, Kaitlyn Zhou, Kayur Patel, Tatsunori Hashimoto, and Michael S Bernstein. 2021. The disagreement deconvolution: Bringing machine learning performance metrics in line with reality. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, pages 1–14.

Robert Gorwa, Reuben Binns, and Christian Katzenbach. 2020. Algorithmic content moderation: Technical and political challenges in the automation of platform governance. *Big Data & Society*, 7(1):2053951719897945.

Tommi Gröndahl, Luca Pajola, Mika Juuti, Mauro Conti, and N Asokan. 2018. All you need is "love": Evading hate speech detection. In *Proceedings of the 11th ACM Workshop on Artificial Intelligence and Security*, pages 2–12. Association for Computing Machinery.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360. Association for Computational Linguistics.

Jessica Guynn. 2019. Facebook while black: Users call it getting zucked, say talking about racism is censored as hate speech. *USA Today*.

Kevin A. Hallgren. 2012. Computing inter-rater reliability for observational data: An overview and tutorial. *Tutorials in Quantitative Methods for Psychology*, 8(1):23–34.

Skyler Hallinan, Alisa Liu, Yejin Choi, and Maarten Sap. 2022. Detoxifying text with marco: Controllable revision with experts and anti-experts. *arXiv preprint arXiv:2212.10543*.

William L. Hamilton, Jure Leskovec, and Dan Jurafsky. 2016a. Cultural shift or linguistic drift? Comparing two computational measures of semantic change. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2116–2121, Austin, Texas. Association for Computational Linguistics.

William L. Hamilton, Jure Leskovec, and Dan Jurafsky. 2016b. Diachronic word embeddings reveal statistical laws of semantic change. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1489–1501, Berlin, Germany. Association for Computational Linguistics.

Andrew F Hayes and Li Cai. 2007. Using heteroskedasticity-consistent standard error estimators in OLS regression: An introduction and software implementation. *Behavior research methods*, 39(4):709–722.

Haibo He and Edwardo A Garcia. 2009. Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering*, 21(9):1263–1284.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2021. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing. *arXiv preprint arXiv:2111.09543*.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2020. Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654*.

Michael A. Hedderich, David Adelani, Dawei Zhu, Jesujoba Alabi, Udia Markus, and Dietrich Klakow. 2020. Transfer learning and distant supervision for multilingual transformer models: A study on African languages. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2580–2591, Online. Association for Computational Linguistics.

Peter Henderson, Jieru Hu, Joshua Romoff, Emma Brunskill, Dan Jurafsky, and Joelle Pineau. 2020. Towards the systematic reporting of the energy and carbon footprints of machine learning. *Journal of Machine Learning Research*, 21(248):1–43.

Alex Hern. 2021. Decoding emojis and defining 'support': Facebook's rules for content revealed. *The Guardian*.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katherine Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Oriol Vinyals, Jack William Rae, and Laurent Sifre. 2022. An empirical analysis of compute-optimal large language model training. In *Advances in Neural Information Processing Systems*.

Valentin Hofmann, Janet Pierrehumbert, and Hinrich Schütze. 2021a. Dynamic contextualized word embeddings. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6970–6984, Online. Association for Computational Linguistics.

Valentin Hofmann, Janet B. Pierrehumbert, and Hinrich Schütze. 2021b. Modeling ideological agenda setting and framing in polarized online groups with graph neural networks and structured sparsity. *CoRR*, abs/2104.08829.

Hossein Hosseini, Sreeram Kannan, Baosen Zhang, and Radha Poovendran. 2017. Deceiving Google's Perspective API built for detecting toxic comments. *arXiv preprint arXiv:1702.08138*.

Dirk Hovy, Taylor Berg-Kirkpatrick, Ashish Vaswani, and Eduard Hovy. 2013. Learning whom to trust with MACE. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1120–1130, Atlanta, Georgia. Association for Computational Linguistics.

Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. 2020. Xtreme: A massively multilingual multi-task benchmark for evaluating cross-lingual generalisation. In *International Conference on Machine Learning*, pages 4411–4421. PMLR.

Kexin Huang, Jaan Altosaar, and Rajesh Ranganath. 2019. ClinicalBERT: Modeling clinical notes and predicting hospital readmission. *CoRR*, abs/1904.05342.

Xiaolei Huang and Michael J. Paul. 2018. Examining temporality in document classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 694–699. Association for Computational Linguistics.

Xiaolei Huang and Michael J. Paul. 2019. Neural temporality adaptation for document classification: Diachronic word embeddings and domain adaptation models. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4113–4123, Florence, Italy. Association for Computational Linguistics.

Chia-Chien Hung, Anne Lauscher, Ivan Vulić, Simone Ponzetto, and Goran Glavaš. 2022. Multi2WOZ: A robust multilingual dataset and conversational pretraining for task-oriented dialog. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3687–3703, Seattle, United States. Association for Computational Linguistics.

Pierre Isabelle, Colin Cherry, and George Foster. 2017. A challenge set approach to evaluating machine translation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2486–2496, Copenhagen, Denmark. Association for Computational Linguistics.

ISO. 2017. Systems and Software Engineering – Vocabulary. Standard, International Organization for Standardisation, Geneva, Switzerland.

Mohit Iyyer, Peter Enns, Jordan Boyd-Graber, and Philip Resnik. 2014. Political ideology detection using recursive neural networks. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1113–1122, Baltimore, Maryland. Association for Computational Linguistics.

Kokil Jaidka, Niyati Chhaya, and Lyle Ungar. 2018. Diachronic degradation of language models: Insights from social media. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 195–200. Association for Computational Linguistics.

Emily Jamison and Iryna Gurevych. 2015. Noise or additional information? leveraging crowdsource annotation item agreement for natural language tasks. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 291–297, Lisbon, Portugal. Association for Computational Linguistics.

Joel Jang, Seonghyeon Ye, Changho Lee, Sohee Yang, Joongbo Shin, Janghoon Han, Gyeonghun Kim, and Minjoon Seo. 2022. TemporalWiki: A lifelong benchmark for training and evaluating ever-evolving language models. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6237–6250, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Jialun Aaron Jiang, Morgan Klaus Scheuerman, Casey Fiesler, and Jed R Brubaker. 2021. Understanding international perceptions of the severity of harmful content online. *PloS one*, 16(8):e0256762.

Xisen Jin, Dejiao Zhang, Henghui Zhu, Wei Xiao, Shang-Wen Li, Xiaokai Wei, Andrew Arnold, and Xiang Ren. 2022. Lifelong pretraining: Continually adapting language models to emerging corpora. In *Proceedings of BigScience Episode #5 – Workshop on Challenges & Perspectives in Creating Large Language Models*, pages 1–16, virtual+Dublin. Association for Computational Linguistics.

Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. The state and fate of linguistic diversity and inclusion in the NLP world. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6282–6293, Online. Association for Computational Linguistics.

Paul Justice. 2006. *Relevant Linguistics*, 2nd edition. Center for the Study of Language and Information, Stanford University.

Sandeepa Kannangara. 2018. Mining Twitter for fine-grained political opinion polarity classification, ideology detection and sarcasm detection. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, pages 751–752.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.

Divyansh Kaushik, Eduard Hovy, and Zachary C. Lipton. 2020. Learning the difference that makes a difference with counterfactually-augmented data. In *Proceedings of the 8th International Conference on Learning Representations*.

Brendan Kennedy, Mohammad Atari, Aida Mostafazadeh Davani, Leigh Yeh, Ali Omrani, Yehsong Kim, Kris Coombs, Shreya Havaldar, Gwenyth Portillo-Wightman, Elaine Gonzalez, Joseph Hoover, Aida Azatian, Alyzeh Hussain, Austin Lara, Gabriel Olmos, Adam Omary, Christina Park, Clarisa Wijaya, Xin Wang, Yong Zhang, and Morteza Dehghani. 2018. The Gab Hate Corpus: A collection of 27k posts annotated for hate speech. *PsyArXiv*.

Brendan Kennedy, Xisen Jin, Aida Mostafazadeh Davani, Morteza Dehghani, and Xiang Ren. 2020. Contextualizing hate speech classifiers with post-hoc explanation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5435–5442, Online. Association for Computational Linguistics.

Kian Kenyon-Dean, Eisha Ahmed, Scott Fujimoto, Jeremy Georges-Filteau, Christopher Glasz, Barleen Kaur, Auguste Lalande, Shruti Bhanderi, Robert Belfer, Nirmal Kanagasabai, Roman Sarrazingendron, Rohit Verma, and Derek Ruths. 2018. Sentiment analysis: It's complicated! 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 1886–1895, New Orleans, Louisiana. Association for Computational Linguistics.

Douwe Kiela, Max Bartolo, Yixin Nie, Divyansh Kaushik, Atticus Geiger, Zhengxuan Wu, Bertie Vidgen, Grusha Prasad, Amanpreet Singh, Pratik Ringshia, Zhiyi Ma, Tristan Thrush, Sebastian Riedel, Zeerak Waseem, Pontus Stenetorp, Robin Jia, Mohit Bansal, Christopher Potts, and Adina Williams. 2021. Dynabench: Rethinking benchmarking in NLP. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4110–4124, Online. Association for Computational Linguistics.

Douwe Kiela, Hamed Firooz, Aravind Mohan, Vedanuj Goswami, Amanpreet Singh, Pratik Ringshia, and Davide Testuggine. 2020. The hateful memes challenge: Detecting hate speech in multimodal memes. *Advances in Neural Information Processing Systems*, 33:2611–2624.

Hannah Kirk, Abeba Birhane, Bertie Vidgen, and Leon Derczynski. 2022a. Handling and presenting harmful text in NLP research. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 497–510, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Hannah Kirk, Bertie Vidgen, and Scott Hale. 2022b. Is more data better? re-thinking the importance of efficiency in abusive language detection with transformers-based active learning. In *Proceedings of the Third Workshop on Threat, Aggression and Cyberbullying (TRAC 2022)*, pages 52–61, Gyeongju, Republic of Korea. Association for Computational Linguistics.

Hannah Kirk, Bertie Vidgen, Paul Rottger, Tristan Thrush, and Scott Hale. 2022c. Hatemoji: A test suite and adversarially-generated dataset for benchmarking and detecting emoji-based hate. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1352–1368, Seattle, United States. Association for Computational Linguistics.

Ben Krause, Emmanuel Kahembwe, Iain Murray, and Steve Renals. 2019. Dynamic evaluation of transformer language models. *CoRR*, abs/1904.08378.

Deepak Kumar, Patrick Gage Kelley, Sunny Consolvo, Joshua Mason, Elie Bursztein, Zakir Durumeric, Kurt Thomas, and Michael Bailey. 2021. Designing toxic content classification for a diversity of perspectives. In *Seventeenth Symposium on Usable Privacy and Security (SOUPS 2021)*, pages 299–318.

Jana Kurrek, Haji Mohammad Saleem, and Derek Ruths. 2020. Towards a comprehensive taxonomy and large-scale annotated corpus for online slur usage. In *Proceedings of the Fourth Workshop on Online Abuse and Harms*, pages 138–149, Online. Association for Computational Linguistics.

Irene Kwok and Yuzhou Wang. 2013. Locate the hate: Detecting tweets against blacks. In *AAAI*.

J. Richard Landis and Gary G. Koch. 1977. The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159.

Craig Langran. 2021. Why was my tweet about football labelled abusive? *BBC News*.

Anne Lauscher, Vinit Ravishankar, Ivan Vulić, and Goran Glavaš. 2020. From zero to hero: On the limitations of zero-shot language transfer with multilingual Transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4483–4499, Online. Association for Computational Linguistics.

Angeliki Lazaridou, Adhiguna Kuncoro, Elena Gribovskaya, Devang Agrawal, Adam Liska, Tayfun Terzi, Mai Gimenez, Cyprien de Masson d'Autume, Sebastian Ruder, Dani Yogatama, Kris Cao, Tomás Kociský, Susannah Young, and Phil Blunsom. 2021. Pitfalls of static language modelling. *CoRR*, abs/2102.01951.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. BioBERT: A pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Alyssa Lees, Vinh Q Tran, Yi Tay, Jeffrey Sorensen, Jai Gupta, Donald Metzler, and Lucy Vasserman. 2022. A new generation of perspective api: Efficient multilingual character-level transformers. *arXiv preprint arXiv:2202.11176*.

João Augusto Leite, Diego Silva, Kalina Bontcheva, and Carolina Scarton. 2020. Toxic language detection in social media for Brazilian Portuguese: New dataset and multilingual analysis. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 914–924, Suzhou, China. Association for Computational Linguistics.

Elisa Leonardelli, Stefano Menini, Alessio Palmero Aprosio, Marco Guerini, and Sara Tonelli. 2021. Agreeing to disagree: Annotating offensive language datasets with annotators' disagreement. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10528–10539, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Hector Levesque, Ernest Davis, and Leora Morgenstern. 2012. The Winograd schema challenge. In *13th International Conference on the Principles of Knowledge Representation and Reasoning*.

Yaobo Liang, Nan Duan, Yeyun Gong, Ning Wu, Fenfei Guo, Weizhen Qi, Ming Gong, Linjun Shou, Daxin Jiang, Guihong Cao, Xiaodong Fan, Ruofei Zhang, Rahul Agrawal, Edward Cui, Sining Wei, Taroon Bharti, Ying Qiao, Jiun-Hung Chen, Winnie Wu, Shuguang Liu, Fan Yang, Daniel Campos, Rangan Majumder, and Ming Zhou. 2020. XGLUE: A new benchmark datasetfor cross-lingual pre-training, understanding and generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6008–6018, Online. Association for Computational Linguistics.

Yu-Hsiang Lin, Chian-Yu Chen, Jean Lee, Zirui Li, Yuyan Zhang, Mengzhou Xia, Shruti Rijhwani, Junxian He, Zhisong Zhang, Xuezhe Ma, Antonios Anastasopoulos, Patrick Littell, and Graham Neubig. 2019. Choosing transfer languages for cross-lingual learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3125–3135, Florence, Italy. Association for Computational Linguistics.

Adam Liska, Tomas Kocisky, Elena Gribovskaya, Tayfun Terzi, Eren Sezener, Devang Agrawal, D'Autume Cyprien De Masson, Tim Scholtes, Manzil Zaheer, Susannah Young, et al. 2022. Streamingqa: A benchmark for adaptation to new knowledge over time in question answering models. In *International Conference on Machine Learning*, pages 13604–13622. PMLR.

Nelson F. Liu, Roy Schwartz, and Noah A. Smith. 2019a. Inoculation by fine-tuning: A method for analyzing challenge datasets. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2171–2179, Minneapolis, Minnesota. Association for Computational Linguistics.

Ruijun Liu, Yuqian Shi, Changjiang Ji, and Ming Jia. 2019b. A survey of sentiment analysis based on transfer learning. *IEEE Access*, 7:85401–85412.

Shuhua Liu and Thomas Forss. 2014. Combining n-gram based similarity analysis with sentiment analysis in web content classification. In *KDIR*, pages 530–537.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019c. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Philipp Lorenz-Spreen, Lisa Oswald, Stephan Lewandowsky, and Ralph Hertwig. 2022. A systematic review of worldwide causal and correlational evidence on digital media and democracy. *Nature human behaviour*, pages 1–28.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *Proceedings of the 7th International Conference on Learning Representations*.

Jan Lukes and Anders Søgaard. 2018. Sentiment analysis under temporal shift. In *Proceedings of the 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 65–71, Brussels, Belgium. Association for Computational Linguistics.

Yiwei Luo, Dallas Card, and Dan Jurafsky. 2020. Detecting stance in media on global warming. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3296–3315, Online. Association for Computational Linguistics.

Rijul Magu and Jiebo Luo. 2018. Determining code words in euphemistic hate speech using word embedding networks. In *Proceedings of the 2nd Workshop on Abusive Language Online (ALW2)*, pages 93–100.

Shervin Malmasi and Marcos Zampieri. 2018. Challenges in discriminating profanity from hate speech. *Journal of Experimental & Theoretical Artificial Intelligence*, 30(2):187–202.

Thomas Mandl, Sandip Modha, Prasenjit Majumder, Daksh Patel, Mohana Dave, Chintak Mandlia, and Aditya Patel. 2019. Overview of the hasoc track at fire 2019: Hate speech and offensive content identification in indo-european languages. In *Proceedings of the 11th Forum for Information Retrieval Evaluation*, pages 14–17.

Marta Marchiori Manerba and Sara Tonelli. 2021. Fine-grained fairness analysis of abusive language detection systems with CheckList. In *Proceedings of the 5th Workshop on Online Abuse and Harms (WOAH 2021)*, pages 81–91, Online. Association for Computational Linguistics.

Nahema Marchal. 2020. The polarizing potential of intergroup affect in online political discussions: Evidence from Reddit r/Politics. *SSRN*.

Delia Marinescu. 2021. Facebook's content moderation language barrier. *New America*.

Todor Markov, Chong Zhang, Sandhini Agarwal, Tyna Eloundou, Teddy Lee, Steven Adler, Angela Jiang, and Lilian Weng. 2022. A holistic approach to undesired content detection in the real world. *arXiv preprint arXiv:2208.03274*.

Louis Martin, Benjamin Muller, Pedro Javier Ortiz Suárez, Yoann Dupont, Laurent Romary, Éric de la Clergerie, Djamé Seddah, and Benoît Sagot. 2020. CamemBERT: a tasty French language model. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7203–7219, Online. Association for Computational Linguistics.

Rebecca Marvin and Tal Linzen. 2018. Targeted syntactic evaluation of language models. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1192–1202, Brussels, Belgium. Association for Computational Linguistics.

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

Mari J Matsuda. 1993. *Words that wound: Critical race theory, assaultive speech, and the first amendment*. Routledge.

Tom McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448, Florence, Italy. Association for Computational Linguistics.

Julia Mendelsohn, Yulia Tsvetkov, and Dan Jurafsky. 2020. A framework for the computational linguistic analysis of dehumanization. *Frontiers in Artificial Intelligence*, 3:55.

Meta. 2022. *Community Standards Enforcement Report Q2 2022*. Meta.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Sandip Modha, Thomas Mandl, Gautam Kishore Shahi, Hiren Madhu, Shrey Satapara, Tharindu Ranasinghe, and Marcos Zampieri. 2021. Overview of the hasoc subtrack at fire 2021: Hate speech and offensive content identification in english and indo-aryan languages and conversational hate speech. In *Forum for Information Retrieval Evaluation*, FIRE 2021, page 1–3, New York, NY, USA. Association for Computing Machinery.

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

Marzieh Mozafari, Reza Farahbakhsh, and Noel Crespi. 2019. A BERT-based transfer learning approach for hate speech detection in online social media. In *International Conference on Complex Networks and Their Applications*, pages 928–940. Springer.

Marzieh Mozafari, Reza Farahbakhsh, and Noël Crespi. 2020. Hate speech detection and racial bias mitigation in social media based on BERT model. *PloS one*, 15(8):e0237861.

Karsten Müller and Carlo Schwarz. 2021. Fanning the flames of hate: Social media and hate crime. *Journal of the European Economic Association*, 19(4):2131–2167.

Aakanksha Naik, Abhilasha Ravichander, Norman Sadeh, Carolyn Rose, and Graham Neubig. 2018. Stress test evaluation for natural language inference. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2340–2353, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Shoichi Naito, Shintaro Sawada, Chihiro Nakagawa, Naoya Inoue, Kenshi Yamaguchi, Iori Shimizu, Farjana Sultana Mim, Keshav Singh, and Kentaro Inui. 2022. TYPIC: A corpus of template-based diagnostic comments on argumentation. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 5916–5928, Marseille, France. European Language Resources Association.

Isar Nejadgholi and Svetlana Kiritchenko. 2020. On cross-dataset generalization in automatic detection of online abuse. In *Proceedings of the Fourth Workshop on Online Abuse and Harms*, pages 173–183, Online. Association for Computational Linguistics.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2020. Adversarial NLI: A new benchmark for natural language understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4885–4901, Online. Association for Computational Linguistics.

Timothy Niven and Hung-Yu Kao. 2019. Probing neural network comprehension of natural language arguments. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4658–4664, Florence, Italy. Association for Computational Linguistics.

Chikashi Nobata, Joel Tetreault, Achint Thomas, Yashar Mehdad, and Yi Chang. 2016. Abusive language detection in online user content. In *Proceedings of the 25th International Conference on World Wide Web*, WWW '16, page 145–153, Republic and Canton of Geneva, CHE. International World Wide Web Conferences Steering Committee.

Curtis Northcutt, Lu Jiang, and Isaac Chuang. 2021. Confident learning: Estimating uncertainty in dataset labels. *Journal of Artificial Intelligence Research*, 70:1373–1411.

Debora Nozza. 2021. Exposing the limits of zero-shot cross-lingual hate speech detection. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 907–914, Online. Association for Computational Linguistics.

Debora Nozza, Federico Bianchi, and Giuseppe Attanasio. 2022. HATE-ITA: Hate speech detection in Italian social media text. In *Proceedings of the Sixth Workshop on Online Abuse and Harms (WOAH)*, pages 252–260, Seattle, Washington (Hybrid). Association for Computational Linguistics.

Debora Nozza, Federico Bianchi, and Dirk Hovy. 2020. What the [MASK]? making sense of language-specific BERT models. *arXiv preprint arXiv:2003.02912*.

OpenAI. 2023. Gpt-4 technical report. *preprint*.

Nedjma Ousidhoum, Zizheng Lin, Hongming Zhang, Yangqiu Song, and Dit-Yan Yeung. 2019. Multilingual and multi-aspect hate speech analysis. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4675–4684, Hong Kong, China. Association for Computational Linguistics.

Nedjma Ousidhoum, Yangqiu Song, and Dit-Yan Yeung. 2020. Comparative Evaluation of Label-Agnostic Selection Bias in Multilingual Hate Speech Datasets. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2532–2542. Association for Computational Linguistics.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Gray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*.

Alexis Palmer, Christine Carr, Melissa Robinson, and Jordan Sanders. 2020. COLD: Annotation scheme and evaluation data set for complex offensive language in English. *Journal for Language Technology and Computational Linguistics*, pages 1–28.

Endang Wahyu Pamungkas, Valerio Basile, and Viviana Patti. 2021. A joint learning approach with knowledge injection for zero-shot cross-lingual hate speech detection. *Information Processing & Management*, 58(4):102544.

Ji Ho Park and Pascale Fung. 2017. One-step and two-step classification for abusive language detection on Twitter. In *Proceedings of the First Workshop on Abusive Language Online*, pages 41–45, Vancouver, BC, Canada. Association for Computational Linguistics.

Ji Ho Park, Jamin Shin, and Pascale Fung. 2018. Reducing gender bias in abusive language detection. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2799–2804, Brussels, Belgium. Association for Computational Linguistics.

Silviu Paun, Bob Carpenter, Jon Chamberlain, Dirk Hovy, Udo Kruschwitz, and Massimo Poesio. 2018. Comparing Bayesian models of annotation. *Transactions of the Association for Computational Linguistics*, 6:571–585.

Ellie Pavlick and Tom Kwiatkowski. 2019. Inherent disagreements in human textual inferences. *Transactions of the Association for Computational Linguistics*, 7:677–694.

John Pavlopoulos, Jeffrey Sorensen, Léo Laugier, and Ion Androutsopoulos. 2021. SemEval-2021 task 5: Toxic spans detection. In *Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)*, pages 59–69, Online. Association for Computational Linguistics.

Andraz Pelicon, Ravi Shekhar, Blaz Skrlj, Matthew Purver, and Senja Pollak. 2021. Investigating cross-lingual training for offensive language detection. *PeerJ Computer Science*, 7:e559. Publisher: PeerJ Inc.

Juan Manuel Pérez, Damián Ariel Furman, Laura Alonso Alemany, and Franco M. Luque. 2022. RoBERTuito: a pre-trained language model for social media text in Spanish. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 7235–7243, Marseille, France. European Language Resources Association.

Billy Perrigo. 2023. Exclusive: Openai used kenyan workers on less than $2 per hour to make chatgpt less toxic. *Time*.

Janet B Pierrehumbert. 2012. Burstiness of verbs and derived nouns. In Diana Santos, Krister Linden, and Wanjiku Ng'ang'a, editors, *Shall We Play the Festschrift Game?*, pages 99–115. Springer.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual BERT? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4996–5001, Florence, Italy. Association for Computational Linguistics.

Georgios K Pitsilis, Heri Ramampiaro, and Helge Langseth. 2018. Detecting offensive language in tweets using deep learning. *arXiv preprint arXiv:1801.04433*.

Barbara Plank. 2022. The "problem" of human label variation: On ground truth in data, modeling and evaluation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10671–10682, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Barbara Plank, Dirk Hovy, and Anders Søgaard. 2014a. Learning part-of-speech taggers with inter-annotator agreement loss. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 742–751, Gothenburg, Sweden. Association for Computational Linguistics.

Barbara Plank, Dirk Hovy, and Anders Søgaard. 2014b. Linguistically debatable or just plain wrong? In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 507–511, Baltimore, Maryland. Association for Computational Linguistics.

Fabio Poletto, Valerio Basile, Manuela Sanguinetti, Cristina Bosco, and Viviana Patti. 2021. Resources and benchmark corpora for hate speech detection: a systematic review. *Language Resources and Evaluation*, 55(2):477–523.

Soujanya Poria, Devamanyu Hazarika, Navonil Majumder, and Rada Mihalcea. 2020. Beneath the tip of the iceberg: Current challenges and new directions in sentiment analysis research. *IEEE Transactions on Affective Computing*.

Vinodkumar Prabhakaran, Aida Mostafazadeh Davani, and Mark Diaz. 2021. On releasing annotator-level labels and information in datasets. In *Proceedings of The Joint 15th Linguistic Annotation Workshop (LAW) and 3rd Designing Meaning Representations (DMR) Workshop*, pages 133–138, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jing Qian, Mai ElSherief, Elizabeth Belding, and William Yang Wang. 2018. Leveraging intra-user and inter-user representation learning for automated hate speech detection. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 118–123, New Orleans, Louisiana. Association for Computational Linguistics.

Inioluwa Deborah Raji, Andrew Smart, Rebecca N. White, Margaret Mitchell, Timnit Gebru, Ben Hutchinson, Jamila Smith-Loud, Daniel Theron, and Parker Barnes. 2020. Closing the ai accountability gap: Defining an end-to-end framework for internal algorithmic auditing. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, FAT* '20, page 33–44, New York, NY, USA. Association for Computing Machinery.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. 2020. Beyond accuracy: Behavioral testing of NLP models with CheckList. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4902–4912, Online. Association for Computational Linguistics.

Paul Röttger, Debora Nozza, Federico Bianchi, and Dirk Hovy. 2022a. Data-efficient strategies for expanding hate speech detection into under-resourced languages. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5674–5691, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Paul Röttger and Janet Pierrehumbert. 2021. Temporal adaptation of BERT and performance on downstream document classification: Insights from social media. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2400–2412, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Paul Röttger, Haitham Seelawi, Debora Nozza, Zeerak Talat, and Bertie Vidgen. 2022b. Multilingual HateCheck: Functional tests for multilingual hate speech detection models. In *Proceedings of the Sixth Workshop on Online Abuse and Harms (WOAH)*, pages 154–169, Seattle, Washington (Hybrid). Association for Computational Linguistics.

Paul Röttger, Bertie Vidgen, Dirk Hovy, and Janet Pierrehumbert. 2022c. Two contrasting data annotation paradigms for subjective NLP tasks. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 175–190, Seattle, United States. Association for Computational Linguistics.

Paul Röttger, Bertie Vidgen, Dong Nguyen, Zeerak Talat, Helen Margetts, and Janet Pierrehumbert. 2021. HateCheck: Functional tests for hate speech detection models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 41–58, Online. Association for Computational Linguistics.

Sebastian Ruder, Noah Constant, Jan Botha, Aditya Siddhant, Orhan Firat, Jinlan Fu, Pengfei Liu, Junjie Hu, Dan Garrette, Graham Neubig, and Melvin Johnson. 2021. XTREME-R: Towards more challenging and nuanced multilingual evaluation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10215–10245, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Maja Rudolph and David Blei. 2018. Dynamic embeddings for language evolution. In *Proceedings of the 2018 World Wide Web Conference*, WWW '18, pages 1003–1011. International World Wide Web Conferences Steering Committee.

Joni Salminen, Hind Almerekhi, Ahmed Mohamed Kamel, Soon-gyo Jung, and Bernard J. Jansen. 2019. Online hate ratings vary by extremes: A statistical analysis. In *Proceedings of the 2019 Conference on Human Information Interaction and Retrieval*, CHIIR '19, page 213–217, New York, NY, USA. Association for Computing Machinery.

Joni Salminen, Hind Almerekhi, Milica Milenković, Soon-gyo Jung, Jisun An, Haewoon Kwak, and Bernard Jansen. 2018. Anatomy of online hate: Developing a taxonomy and machine learning models for identifying and classifying hate in online news media. *Proceedings of the 12th International AAAI Conference on Web and Social Media*, 1(1):330–339.

Mattia Samory, Indira Sen, Julian Kohne, Fabian Floeck, and Claudia Wagner. 2021. "call me sexist, but..." : Revisiting sexism detection using psychological scales and adversarial samples. *Proceedings of the International AAAI Conference on Web and Social Media,*.

Manuela Sanguinetti, Gloria Comandini, Elisa Di Nuovo, Simona Frenda, Marco Stranisci, Cristina Bosco, Tommaso Caselli, Viviana Patti, and Irene Russo. 2020. Haspeede 2 @ EVALITA2020: overview of the EVALITA 2020 hate speech detection task. In *Proceedings of the Seventh Evaluation Campaign of Natural Language Processing and Speech Tools for Italian. Final Workshop (EVALITA 2020), Online event, December 17th, 2020*, volume 2765 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Maarten Sap, Dallas Card, Saadia Gabriel, Yejin Choi, and Noah A. Smith. 2019. The risk of racial bias in hate speech detection. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1668–1678, Florence, Italy. Association for Computational Linguistics.

Maarten Sap, Saadia Gabriel, Lianhui Qin, Dan Jurafsky, Noah A. Smith, and Yejin Choi. 2020. Social bias frames: Reasoning about social and power implications of language. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5477–5490, Online. Association for Computational Linguistics.

Maarten Sap, Swabha Swayamdipta, Laura Vianna, Xuhui Zhou, Yejin Choi, and Noah A. Smith. 2022. Annotators with attitudes: How annotator beliefs and identities bias toxic language detection. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5884–5906, Seattle, United States. Association for Computational Linguistics.

Anna Schmidt and Michael Wiegand. 2017. A survey on hate speech detection using natural language processing. In *Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media*, pages 1–10, Valencia, Spain. Association for Computational Linguistics.

Mark Scott. 2021. Facebook did little to moderate posts in the world's most violent countries. *Politico*.

Rico Sennrich. 2017. How grammatical is character-level neural machine translation? Assessing MT quality with contrastive translation pairs. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 376–382, Valencia, Spain. Association for Computational Linguistics.

Deven Santosh Shah, H. Andrew Schwartz, and Dirk Hovy. 2020. Predictive biases in natural language processing models: A conceptual framework and overview. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5248–5264, Online. Association for Computational Linguistics.

Shubham Sharma, Yunfeng Zhang, Jesús M. Ríos Aliaga, Djallel Bouneffouf, Vinod Muthusamy, and Kush R. Varshney. 2020. Data augmentation for discrimination prevention and bias disambiguation. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*, AIES '20, page 358–364, New York, NY, USA. Association for Computing Machinery.

Tom Simonite. 2021. Facebook is everywhere; its moderation is nowhere close. *Wired*.

Hajung Sohn and Hyunju Lee. 2019. Mc-bert4hate: Hate speech detection using multi-channel bert for different languages and translations. In *2019 International Conference on Data Mining Workshops (ICDMW)*, pages 551–559. IEEE.

Fábio Souza, Rodrigo Nogueira, and Roberto Lotufo. 2020. BERTimbau: Pretrained BERT models for Brazilian Portuguese. In *Intelligent Systems*, pages 403–417, Cham. Springer International Publishing.

Lukas Stappen, Fabian Brunn, and Björn W. Schuller. 2020. Cross-lingual zero- and few-shot hate speech detection utilising frozen transformer language models and AXEL. *CoRR*, abs/2004.13850.

Statista. 2019. Reddit: Traffic by country. *Statista*.

Emma Strubell, Ananya Ganesh, and Andrew McCallum. 2019. Energy and policy considerations for deep learning in NLP. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3645–3650, Florence, Italy. Association for Computational Linguistics.

Hongjin Su, Jungo Kasai, Chen Henry Wu, Weijia Shi, Tianlu Wang, Jiayi Xin, Rui Zhang, Mari Ostendorf, Luke Zettlemoyer, Noah A. Smith, and Tao Yu. 2023. Selective annotation makes language models better few-shot learners. In *The Eleventh International Conference on Learning Representations*.

Shardul Suryawanshi and Bharathi Raja Chakravarthi. 2021. Findings of the shared task on troll meme classification in tamil. In *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*, pages 126–132.

Zeerak Talat. 2016. Are you a racist or am I seeing things? Annotator influence on hate speech detection on Twitter. In *Proceedings of the First Workshop on NLP and Computational Social Science*, pages 138–142, Austin, Texas. Association for Computational Linguistics.

Zeerak Talat, Thomas Davidson, Dana Warmsley, and Ingmar Weber. 2017. Understanding abuse: A typology of abusive language detection subtasks. In *Proceedings of the First Workshop on Abusive Language Online*, pages 78–84, Vancouver, BC, Canada. Association for Computational Linguistics.

Zeerak Talat and Dirk Hovy. 2016. 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. Association for Computational Linguistics.

Zeerak Talat, James Thorne, and Joachim Bingel. 2018. Bridging the gaps: Multi-task learning for domain transfer of hate speech detection. In Jennifer Golbeck, editor, *Online Harassment*. Springer.

Jacques P Thiroux and Keith W Krasemann. 2015. *Ethics: Theory and Practice*, 11th edition. Pearson.

Kshitiz Tiwari, Shuhan Yuan, and Lu Zhang. 2022. Robust hate speech detection via mitigating spurious correlations. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 51–56, Online only. Association for Computational Linguistics.

Thanh Tran, Yifan Hu, Changwei Hu, Kevin Yen, Fei Tan, Kyumin Lee, and Se Rim Park. 2020. HABERTOR: An efficient and effective deep hatespeech detector. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7486–7502, Online. Association for Computational Linguistics.

Adam Tsakalidis and Maria Liakata. 2020. Sequential modelling of the evolution of word representations for semantic change detection. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8485–8497, Online. Association for Computational Linguistics.

Stéphan Tulkens, Lisa Hilte, Elise Lodewyckx, Ben Verhoeven, and Walter Daelemans. 2016. A dictionary-based approach to racism detection in dutch social media. *arXiv preprint arXiv:1608.08738*.

UAE. 2022. Anti-discrimination / anti-hatred law. *United Arab Emirates*.

Alexandra Uma, Tommaso Fornaciari, Anca Dumitrache, Tristan Miller, Jon Chamberlain, Barbara Plank, Edwin Simpson, and Massimo Poesio. 2021. SemEval-2021 task 12: Learning with disagreements. In *Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)*, pages 338–347, Online. Association for Computational Linguistics.

Alexandra Uma, Tommaso Fornaciari, Dirk Hovy, Silviu Paun, Barbara Plank, and Massimo Poesio. 2020. A case for soft loss functions. *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, 8(1):173–177.

Betty van Aken, Julian Risch, Ralf Krestel, and Alexander Löser. 2018. Challenges for toxic comment classification: An in-depth error analysis. In *Proceedings of the 2nd Workshop on Abusive Language Online (ALW2)*, pages 33–42, Brussels, Belgium. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Bertie Vidgen, Emily Burden, and Helen Margetts. 2021a. Understanding online hate: Vsp regulation and the broader context. *Turing Institute*.

Bertie Vidgen and Leon Derczynski. 2020. Directions in abusive language training data, a systematic review: Garbage in, garbage out. *PLOS ONE*, 15(12):e0243300.

Bertie Vidgen, Scott Hale, Ella Guest, Helen Margetts, David Broniatowski, Zeerak Talat, Austin Botelho, Matthew Hall, and Rebekah Tromble. 2020. Detecting East Asian prejudice on social media. In *Proceedings of the Fourth Workshop on Online Abuse and Harms*, pages 162–172, Online. Association for Computational Linguistics.

Bertie Vidgen, Alex Harris, Dong Nguyen, Rebekah Tromble, Scott Hale, and Helen Margetts. 2019. Challenges and frontiers in abusive content detection. In *Proceedings of the Third Workshop on Abusive Language Online*, pages 80–93, Florence, Italy. Association for Computational Linguistics.

Bertie Vidgen, Dong Nguyen, Helen Margetts, Patricia Rossini, and Rebekah Tromble. 2021b. 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. Association for Computational Linguistics.

Bertie Vidgen, Tristan Thrush, Zeerak Talat, and Douwe Kiela. 2021c. Learning from the worst: Dynamically generated datasets to improve online hate detection. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1667–1682, Online. Association for Computational Linguistics.

Jeremy Waldron. 2012. *The harm in hate speech*. Harvard University Press.

Cindy Wang and Michele Banko. 2021. Practical Transformer-based Multilingual Text Classification. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Industry Papers*, pages 121–129, Online. Association for Computational Linguistics.

Xinyi Wang, Sebastian Ruder, and Graham Neubig. 2022. Expanding pretrained models to thousands more languages via lexicon-based adaptation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 863–877, Dublin, Ireland. Association for Computational Linguistics.

William Warner and Julia Hirschberg. 2012. Detecting hate speech on the World Wide Web. In *Proceedings of the Second Workshop on Language in Social Media*, pages 19–26, Montréal, Canada. Association for Computational Linguistics.

Alex Warstadt, Alicia Parrish, Haokun Liu, Anhad Mohananey, Wei Peng, Sheng-Fu Wang, and Samuel R. Bowman. 2020. BLiMP: The benchmark of linguistic minimal pairs for English. *Transactions of the Association for Computational Linguistics*, 8:377–392.

Michael Wiegand, Josef Ruppenhofer, and Thomas Kleinbauer. 2019. Detection of abusive language: The problem of biased datasets. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 602–608, Minneapolis, Minnesota. Association for Computational Linguistics.

Matthew L Williams, Pete Burnap, Amir Javed, Han Liu, and Sefa Ozalp. 2020. Hate in the machine: Anti-black and anti-muslim social media posts as predictors of offline racially and religiously aggravated crime. *The British Journal of Criminology*, 60(1):93–117.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Shijie Wu and Mark Dredze. 2019. Beto, bentz, becas: The surprising cross-lingual effectiveness of BERT. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 833–844, Hong Kong, China. Association for Computational Linguistics.

Tongshuang Wu, Marco Tulio Ribeiro, Jeffrey Heer, and Daniel Weld. 2019. Errudite: Scalable, reproducible, and testable error analysis. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 747–763, Florence, Italy. Association for Computational Linguistics.

Ellery Wulczyn, Nithum Thain, and Lucas Dixon. 2017. Ex machina: Personal attacks seen at scale. In *Proceedings of the 26th International Conference on World Wide Web*, pages 1391–1399.

Zhiping Xiao, Weiping Song, Haoyan Xu, Zhicheng Ren, and Yizhou Sun. 2020. Timme: Twitter ideology-detection via multi-task multi-relational embedding. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2258–2268.

Fan Yang, Xiaochang Peng, Gargi Ghosh, Reshef Shilon, Hao Ma, Eider Moore, and Goran Predovic. 2019. Exploring deep multimodal fusion of text and photo for hate speech classification. In *Proceedings of the Third Workshop on Abusive Language Online*, pages 11–18, Florence, Italy. Association for Computational Linguistics.

Wenjie Yin and Arkaitz Zubiaga. 2021. Towards generalisable hate speech detection: a review on obstacles and solutions. *PeerJ Computer Science*, 7:e598.

Annie Zaenen. 2006. Mark-up barking up the wrong tree. *Comput. Linguist.*, 32(4):577–580.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019a. Predicting the type and target of offensive posts in social media. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1415–1420, Minneapolis, Minnesota. Association for Computational Linguistics.

Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019b. SemEval-2019 task 6: Identifying and categorizing offensive language in social media (OffensEval). In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 75–86, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Philine Zeinert, Nanna Inie, and Leon Derczynski. 2021. Annotating online misogyny. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3181–3197, Online. Association for Computational Linguistics.

Jing Zhang, Victor S Sheng, Tao Li, and Xindong Wu. 2017. Improving crowdsourced label quality using noise correction. *IEEE transactions on neural networks and learning systems*, 29(5):1675–1688.

Ziqi Zhang and Lei Luo. 2019. Hate speech detection: A solved problem? The challenging case of long tail on Twitter. *Semantic Web*, 10(5):925–945.

Ziqi Zhang, David Robinson, and Jonathan Tepper. 2018. Detecting hate speech on Twitter using a convolution-GRU based deep neural network. In *European Semantic Web conference*, pages 745–760. Springer.

Jieyu Zhao, Subhabrata Mukherjee, Saghar Hosseini, Kai-Wei Chang, and Ahmed Hassan Awadallah. 2020. Gender bias in multilingual embeddings and cross-lingual transfer. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2896–2907, Online. Association for Computational Linguistics.

Mengjie Zhao, Yi Zhu, Ehsan Shareghi, Ivan Vulić, Roi Reichart, Anna Korhonen, and Hinrich Schütze. 2021. A closer look at few-shot crosslingual transfer: The choice of shots matters. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5751–5767, Online. Association for Computational Linguistics.

Lucia Zheng, Neel Guha, Brandon R. Anderson, Peter Henderson, and Daniel E. Ho. 2021. When does pretraining help? Assessing self-supervised learning for law and the CaseHOLD dataset of 53,000+ legal holdings. In *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Law*, ICAIL '21, page 159–168, New York, NY, USA. Association for Computing Machinery.

Xiang Zhou, Yixin Nie, Hao Tan, and Mohit Bansal. 2020. The curse of performance instability in analysis datasets: Consequences, source, and suggestions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8215–8228, Online. Association for Computational Linguistics.

Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the IEEE international conference on computer vision*, pages 19–27.

Haris Bin Zia, Ignacio Castro, Arkaitz Zubiaga, and Gareth Tyson. 2022. Improving zero-shot cross-lingual hate speech detection with pseudo-label fine-tuning of transformer language models. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 16, pages 1435–1439.