Master Thesis

What's in a phrase? Identifying implicit hate with generative AI

Victoria Im

Supervisor Isa Maks
2^{nd} reader L.G. de Passos Morgado da Costa

a thesis submitted in fulfillment of the requirements for the degree of

MA Linguistics
(Human Language Technology)

Vrije Universiteit Amsterdam

Computational Linguistics and Text-Mining Lab
Department of Language and Communication
Faculty of Humanities

Date August 14, 2025
Student number 2852317

Abstract

This study investigates the generative model Qwen's performance in implicit hate speech detection and compares its performance with the masked language model BERT. To improve the generative model's performance, we implemented some of the prompting methods described by Han and Tang (2022) and Guo et al. (2023) and incorporated finer implicit class labels into the prompts.

The study describes how different prompt methods (incorporation of hate speech definitions, training examples, finer labels, etc.) impact the Qwen model's performance. The best-performing prompt for our experiments is an instruction to classify over three major labels (*explicit hate*, *implicit hate*, *not hate*) and consider implicit class labels in the few-shot setting with eight training examples per major label.

The generative model Qwen did not outperform the BERT model. However, the implementation of various prompting strategies helped to improve its performance.

Qualitative error analysis revealed that the generative model Qwen struggled with the identification of subtle language and linguistic phenomena such as irony and sarcasm. The model in particular struggled to distinguish instances where sentences labeled as *implicit hate* were predicted as *explicit hate* because such sentences contained highly aggressive, dehumanizing, and inciting to violence language. This suggests a misalignment between the model's learned representations and the labeling criteria of ElSherief et al. (2021), with the model tending to classify *implicit hate* sentences as *explicit hate* based on its prior knowledge.

The code used for this thesis can be found at: https://github.com/Victoria-842/VU-MA-Thesis.git

Keywords: Hate Speech Detection, Implicit Hate, Text Mining, Natural Language Processing, Machine Learning, Artificial Intelligence, Generative Model, Masked Language Model, Prompting

Declaration of Authorship

I, author, declare that this thesis, titled *What's in a phrase? Identifying implicit hate with generative AI* and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a degree degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Date: 27.06.2025

Signed: 

Acknowledgments

I would like to express my gratitude to my supervisor Isa Maks for guidance and moral support throughout this short but intense part of the MA program. I would like to express my gratitude to the CLTL department of the VU University for the knowledge shared throughout this program.

I would like to thank Martha Koukourikou, Elisabetta Dentico, Matt Mathews, Melina Paxinou, Manya Walavalkar, your support and companionship made this journey truly memorable and fun.

Finally, I thank my family and friends for lending me an ear and a shoulder during this challenging year.

List of Figures

List of Tables

Contents

Chapter 1

Introduction

1.1 Problem description

Social media platforms are often envisioned as virtual spaces that facilitate meaningful dialogue and the exchange of diverse perspectives in global communities. However, this potential is frequently undermined by antisocial and abusive behaviors, including harassment, hate speech, trolling, and other forms of online aggression (Jurgens et al., 2019). With the development of social media, the automated detection of hate speech became one of the most urgent tasks in NLP, and various methods were developed to detect specific forms of abusive content and support automated content moderation (Caselli et al., 2020). However, these efforts focused predominantly on overt abuse, often overlooking the more nuanced and diverse manifestations of hate speech (Jurgens et al., 2019). Meanwhile, hate speech is often subtle and implicit. Despite its subtlety, such abuse can be just as emotionally damaging as overt expressions of hate (Sue, 2010).

The challenges associated with detecting implicit hate speech stem from the nature of the language used in implicit hate. While explicit hate typically relies on identifiable slurs or overtly hateful expressions that are relatively straightforward to detect, implicit hate often employs more subtle linguistic devices such as metaphor, irony, and sarcasm (Ocampo et al., 2023). Implicit hate often relies on contextual cues, which may be embedded in surrounding text, cultural references, or symbols (Lei et al., 2017). Therefore, the interpretation of implicit hate frequently requires contextual or topical knowledge, further complicating its detection.

Ocampo et al. (2023) demonstrated that state-of-the-art neural networks sufficiently detected explicit hate speech, but struggled to identify implicit and subtle content. This underscores that the detection of implicit hate speech remains an unsolved problem.

Among the challenges surrounding hate speech research, the definition of hate speech clearly and universally within the field of NLP continues to be a persistent issue. Researchers widely recognize that the lack of clarity around the concept obstructs systematic research and that a more precise definition would greatly enhance future research efforts in this domain (MacAvaney, 2019). According to Nockleby (2000), explicit hate speech is a communication that disparages a person or a group on the basis of certain characteristics such as race, color, ethnicity, gender, religion, etc. ElSherief et al. (2021) identifies that implicit hate speech goes beyond word-related meaning, uses sarcasm, metaphor, and other forms of figurative language to disparage a protected group or individual, or to convey prejudicial and harmful views about them.

The subtle, nuanced nature and diversity of implicit hate also present significant challenges for annotation (ElSherief et al., 2021). Often, annotator disagreements appear because of difficulties in distinguishing between negative but not necessarily hateful messages and implicitly abusive ones; and determining whether a message is abusive must take into account

the context in which it occurred, rather than evaluating the message in isolation (Caselli et al., 2020).

Accurately interpreting implicit hate speech often demands a high level of cultural and political awareness from the annotator. Therefore, creating comprehensive annotation guidelines that fully capture the diverse contexts in which hate speech can be subtly conveyed remains a significant challenge (MacAvaney, 2019). Consider a sentence:

"Uh white lives matter?"

The sentence was labeled as *implicit hate* in the dataset of ElSherief et al. (2021), but what makes it implicitly hateful? To be able to label this sentence as *implicit hate*, an annotator should possess the knowledge about political movements in the U.S., a linguistic sense that the filler word "uh" most probably expresses sarcasm and mockery.

In this work, we address the challenge of detecting implicit hate speech, which often requires contextual understanding, by leveraging generative models and prompt-based approaches. The data set on which we test our approaches is the data set with implicit class labels developed by ElSherief et al. (2021) specifically for the implicit hate detection.

1.2 Research objectives

In this thesis, we investigate the performance of the masked language model BERT and the generative model Qwen on the ElSherief et al. (2021) dataset with seven implicit classes. We experiment with some of the prompting methods suggested by Han and Tang (2022) and Guo et al. (2023), and explore how various prompting methods and the incorporation of finer labels influence the performance of the generative model.

Research questions:

1) Can generative model combined with prompting techniques outperform masked language models like BERT in detecting implicit hate speech?

2) What prompting techniques are the most effective for improving the detection of implicit hate speech?

To address these questions, we selected the implicit hate dataset, which was specifically designed for the detection of implicit hate speech (ElSherief et al., 2021). Using this dataset, we conducted a series of experiments aimed at answering the following subquestions.

Sub-questions

1a) What is the performance of BERT on this dataset?

1b) What is the performance of the generative model on this dataset?

2a) How do different prompting techniques affect the results for implicit hate speech detection?

2b) How does incorporating external knowledge about hate speech into the prompts impact performance on implicit hate speech detection?

Generative models are trained on large-scale corpora, allowing them to develop a broad and nuanced understanding of human language. As noted by Guo et al. (2023), this linguistic knowledge contributes to the advanced contextual understanding of human language by generative models and can be effectively leveraged through prompting strategies. Han and Tang (2022) argue that providing a sufficient number of training examples, along with informative task descriptions that include higher-level labels, enhances the performance of generative models. Our working hypothesis builds on both statements, proposing that prompts enriched with additional information, such as training examples and higher-level labels, can effectively leverage the pre-existing knowledge of human language of the generative model to improve performance in implicit hate detection.

1.3 Approach

As a baseline for our research, we used the results of the masked language model. Masked language models have become foundational in natural language processing (NLP) due to their powerful ability to capture complex language patterns and long-range dependencies in text (Devlin et al., 2019). Among such models, BERT (Bidirectional Encoder Representations from Transformers) has emerged as a standard baseline model in NLP research. In the domain of implicit detection of hate speech, which is challenging due to subtle context-dependent language cues, the BERT results are frequently used as a benchmark (Caselli et al., 2021; ElSherief et al., 2021; Ocampo et al., 2023). Researchers often compare the results of newer models or experimental architectures with the performance of BERT to evaluate the performance in detecting implicit hate (ElSherief et al., 2021).

The BERT model is often used as a comparison of the results of experiments with generative models for the detection of hate speech (Guo et al., 2023). We investigate what masked language model performance in implicit hate recognition is and compare it with the generative model performance.

The rapid advancement of generative models has contributed to their growing popularity, not only among users but also as a prominent subject of academic research, and among the various areas of interest, prompt engineering has emerged as a field of study itself (Han and Tang, 2022). Trained on vast and diverse corpora Large Language Models (LLMs) demonstrated the ability to perform NLP tasks based on textual instructions or just a few examples (Brown et al., 2020). It seems logical to test these models on the detection of hate speech. Prompt engineering is the strategic crafting of queries or instructions to guide the behavior of generative models toward the desired outcomes (Guo et al., 2023). This technique is particularly relevant in scenarios where models are expected to perform tasks without explicit fine-tuning, leveraging their zero-shot or few-shot learning capabilities (Brown et al., 2020). To investigate how various prompting methods impact the performance of a generative model, we will use the Qwen2-7B-Instruct model.

The Qwen2-7B-Instruct model with 7 billion parameters has been effectively optimized for a wide range of language and reasoning tasks and outperformed previously released state-of-the-art (SOTA) models of similar size in most benchmark datasets (Yang et al., 2024).

1.4 Thesis structure

Chapter 2 focuses on the few existing challenges related to the research of *implicit hate*: definition of hate speech, annotation, and prompting methods. Chapter 3 describes the data we used for the research. Chapter 4 describes the experimental setup: model parameters, prompt strategies, and modifications. Chapter 5 discusses all models' results. Chapter 6 describes the error analysis of the best-performing prompt. Chapter 7 summarizes the main findings and discusses limitations and future work.

Chapter 2

Related work

2.1 Overview of implicit hate data sets

Research on hate speech has predominantly focused on explicit hate speech, as it is relatively straightforward to identify using hate lexicons with explicitly hateful words (ElSherief et al., 2021). In contrast, research on implicit hate detection is comparatively underdeveloped due to the lack of implicit hate datasets. Few studies attempted to develop English language data sets that would cover implicit hate to varying degrees.

The Gab Hate Corpus (GHC) contains 27,665 posts from the social network service gab.com (Kennedy et al., 2018). Each post was annotated for the presence of "hate-based rhetoric" including a classification that differentiates between implicit and explicit rhetoric, evaluating the "framing" effects of a post, with implicit rhetoric defined as an invocation of derogatory beliefs, sentiments, or threats that are accessible through shared cultural knowledge (Kennedy et al., 2018). The corpus development aimed to provide an integrated approach to hate speech by improving understanding of hate groups and hateful behaviors, detection of hate speech, and developing policies (Kennedy et al., 2018).

SemEval Task 10 on the Explainable Detection of Online Sexism (EDOS) operates with a data set of 20,000 social media comments that were annotated with fine-grained labels that distinguish between various characteristics, including "animosity," which represents implicit or subtle sexism, stereotypes, or descriptive statements (Kirk et al., 2023). The goal of the task was to develop a taxonomy of sexist content and to enable automated, non-binary detection of such content (Kirk et al., 2023).

Shared Task on Aggression Identification organized as part of the First Workshop on Trolling, Aggression, and Cyberbullying (TRAC - 1) at COLING 2018, aimed to develop a classifier that could discriminate between overtly aggressive, covertly aggressive, and non-aggressive texts (Kumar, 2018). The task data set consisted of 15,000 Facebook posts and comments annotated with aggression labels in Hindi and English (Kumar, 2018).

Caselli et al. (2020) re-annotated existing OLID/OffensEval data set (Zampieri et al., 2019) with markers of the degree of explicitness. The data set contains 14,100 English tweets and was originally labeled offensive or not, targeted or not, and also had a label of a type of target (*see Section* 2.3). The authors aimed to address some of the existing issues in the annotation of offensive and abusive language (e.g., explicitness of the message, presence of a target, need of context, and interaction across different phenomena) (Zampieri et al., 2019).

Ocampo et al. (2023) re-annotated the same OLID/OffensEval data set (Zampieri et al., 2019) with three-layer annotation (HS/non HS, Explicit/Implicit, and Subtle/Non Subtle) and 18 typical properties of implicit language (*see Section* 2.3). The focus of the study was on the development of a data set that could help an automatic system detect various forms of hate speech (Ocampo et al., 2023).

ElSherief et al. (2021) developed an implicit hate corpus that contains 21,480 Twitter

posts from U.S. hate groups and contains labels of various implicit classes (*see Section* 3.1). The work focused on developing a benchmark for the task of implicit hate speech detection (ElSherief et al., 2021).

The scarcity of annotated datasets specifically targeting implicit hate limits the development and training of models for automated hate speech detection. Implicit hate often relies on shared cultural knowledge, stereotypes, or indirect framing of posts and can appear in various forms: subtle sexism as in the EDOS dataset (Kirk et al., 2023); covert aggression as in TRAC-1 dataset (Kumar, 2018). These variations make it difficult to create universal annotation guidelines and, consequently, a comprehensive data set.

Apart from a broad variety of implicit hate speech, there are challenges in annotation and inter-annotator agreement. Researchers attempt to re-annotate existing datasets to capture implicit/explicit distinctions and other nuanced properties (e.g., subtlety, extent of explicitness) (Caselli et al., 2020). Annotation often requires multiple layers of labeling (HS/non-HS, explicit/implicit, subtle/non-subtle, and various implicit linguistic properties) (Ocampo et al., 2023).

2.2 Definition of hate speech

The United Nations defines hate speech as any discriminatory communication that targets a person or a group of people "based on who they are, in other words, based on their religion, ethnicity, nationality, race, color, descent, gender, or other identity factor" (United Nations, 2025a). At the same time, the United Nations also acknowledges that there is no formal definition of hate speech under international human rights law, and it is still "in discussion" (United Nations, 2025b). "Hate speech" is not defined as a separate category in international human rights law and can only be restricted if it contains "incitement to discrimination, hostility or violence" (United Nations, 2025b). The task of establishing a clear and universally accepted definition of hate speech within the field of NLP remains an ongoing challenge. Scholars generally acknowledge that the ambiguity surrounding the concept hinders systematic research and that a more precise definition would significantly improve future studies in this area (MacAvaney, 2019).

In the absence of a unifying definition of hate speech, researchers refer to each other's work, defining hate speech in slightly different terms but overall determining hate speech as abusive, insulting, offensive, or threatening and targeting specific groups based on certain characteristics (Ghosh et al., 2023). For explicit hate speech, we adopt the definition provided by Nockleby (2000) because first, it specifically explains a hate speech as attacking an individual and a group; and second, because it is close to the definition of the main target groups of hate speech adopted by the UN. Nockleby (2000) describes hate speech as "any communication that disparages a person or a group on the basis of certain characteristics such as race, color, ethnicity, gender, sexual orientation, nationality, religion, or other characteristics." Furthermore, in our study, we use the dataset provided in the investigation of implicit hate speech by ElSherief et al. (2021). Therefore, we also refer to the implicit hate speech definition utilized in this research, which defines implicit hate speech as "a coded or indirect language such as sarcasm, metaphor, and circumlocution used to disparage a protected group or individual, or to convey prejudicial and harmful views about them." (ElSherief et al., 2021).

2.3 Hate speech annotation

Due to the lack of a clear and widely accepted definition of hate speech, the annotation of hate speech also faces serious challenges. In the absence of a universally accepted definition of hate

speech, annotation practices vary significantly, resulting in datasets that reflect divergent interpretations and different degrees of inter-annotator agreement.

Different definitions of hate speech often lead to contradictory annotation guidelines, and Caselli et al. (2020) address this issue by identifying two basic principles of annotation guidelines: identifying whether a message is directed at a specific target and determining the degree to which a message is clearly recognized as abusive. Directing attention to these objectives helps to differentiate hate speech from offensive language and other social media phenomena; it can also help to create a clearer definition of hate speech (Caselli et al., 2020). The study also aimed to differentiate explicit abuse from implicit one; thus, the authors proposed a three-way annotation with three labels (explicit (abuse), implicit (abuse), not (abusive)) instead of the commonly used binary one. Annotators followed various guidelines to identify whether a message was not abusive (e.g., an abusive message was used within quotes); explicitly abusive (e.g., imperative clause); implicitly abusive (e.g., sarcasm, irony, and rhetorical question) (Caselli et al., 2020). Analysis of disagreements between annotators concluded that negative messages and implicitly abusive messages were often difficult to distinguish, for example, two annotators labeled the same message as implicit abuse and not abusive: "@USER @USER I believe gun control should consist of guarding your firearms from theft and children" [1]. Other linguistic phenomena, such as borderline profanity, slur, irony, sarcasm, also caused many disagreements in many cases due to the missing context (Caselli et al., 2020). One of the conclusions of this investigation identified that the decision on whether a message was abusive must be considered within the context where the message occurred and not in isolation (Caselli et al., 2020).

Various studies have been conducted on different types of hate speech with researchers focusing on offensive (Caselli et al., 2020), abusive (Vidgen, 2021), toxic (Sahoo Nihar, 2022) speech. One of the problems with annotations is that the annotation task can be relatively straightforward for explicit hate speech but more unclear and confusing for implicit hate speech, and annotators should be able to catch the indirect meaning of topic-specific speech (Waseem, 2017). Implicit hate speech is particularly difficult to identify as it often relies on contextual cues, which may be embedded in surrounding text, cultural references, or symbols (Gao, 2017). A study conducted by Ross et al. (2016) concludes that the task of hate speech classification should not be a binary classification, but rather a prediction of the extent to which a message is hateful. The authors suggest how characteristics according to which a text can be marked as hateful can be useful for hate speech detection (Ross et al., 2016). It is a difficult task to clearly define if there is hate speech in the text, and annotators should receive detailed instructions; therefore, attaining high inter-annotator agreement remains challenging, underscoring the need for a more nuanced and fine-grained classification of hate speech (Ross et al., 2016).

An effective interpretation of implicit hate speech often requires that the annotator has a high degree of cultural and political awareness. As a result, it remains challenging (if not impossible) to develop comprehensive annotation guidelines that account for the wide range of contexts in which hate speech may be implicitly expressed. The annotation of implicit hate speech requires a nuanced understanding of the specific domain in which it occurs, whether in political discourse, commentary on economic conditions, or other areas of public life, underscoring the importance of contextual and subject matter expertise in the annotation process (MacAvaney, 2019). The speaker's objective and context are the focus of the implicit hate identification (MacAvaney, 2019). Implicit hate speech uses paraphrasing, metaphors, satire, and stereotypes (Lei et al., 2017).

Consider examples:

(1) "It is driven by Jews carried out by Somalis and Arabs. The issue is keeping Europe

[1] All data examples in this thesis are presented as originally recorded, with spelling and grammar retained from the source.

white."

(2) "I lived there it was fine, you are just a scared white dude."

Example (1) is easily identifiable as explicit hate speech, because it contains clearly hateful words or phrases (Lei et al., 2017). Explicit hate speech uses words in its direct meaning (Ocampo et al., 2023). Example (2) contains "coded or indirect language that disparages a person" (ElSherief et al., 2021), it is not obvious as explicit hate speech in the first example.

Ocampo et al. (2023) argued that the re-annotation implemented by Caselli et al. (2020) mainly used explicit hate speech examples because data filtering mostly relied on keywords, and it would be more accurate to collect data from communities that were potentially susceptible to engaging in hate speech (e.g., implicit hate corpus by ElSherief et al. (2021)). They created a corpus that covered "18 typical properties of implicitness" (e.g. irony, sarcasm, black humor, metaphor, etc.) and re-annotated seven existing datasets with labels "HS/non HS, Explicit/Implicit and Subtle/Non Subtle" and with 18 fine-grained labels of implicit hate speech (e.g. "irony", "exaggeration", "metaphor", "rhetorical question", etc.). The research aimed to investigate various types of implicit hate speech and subtle hate speech. The authors followed the definition of implicit hate speech suggested by ElSherief et al. (2021): implicit hate has an indirect nature; it "goes beyond word-related meaning"; meanwhile, subtle HS follows the literal meaning of words (Ocampo et al., 2023). The annotation was performed by sampling 100 messages from each of the seven data sources; four graduate-level annotators with a background in linguistics conducted the annotations (Ocampo et al., 2023). The inter-annotator Agreement reached Cohen's $\varkappa$=0.793 (binary annotation Explicit/Implicit) and 0.730 for the subtlety layer (binary annotation Subtle/Non-Subtle) (Ocampo et al., 2023). Most disagreements were mainly caused by the intertwined nature of subtlety.

2.4 BERT and HateBERT models for hate speech detection

The BERT masked language model has established itself as a widely adopted baseline for a broad range of NLP tasks, frequently serving as a pre-trained foundation for downstream fine-tuning. It is widely used in the research of implicit hate speech detection, in particular in the works mentioned above by ElSherief et al. (2021); Caselli et al. (2020) (*see Section* 2.3). Therefore, in our study, we first perform experiments with the BERT model and establish its results as a baseline.

BERT (Bidirectional Encoder Representations from Transformers), developed by Google, is a pre-trained language model that captures bidirectional contextual representations of text. Its ability to simultaneously consider the left and right contexts enables a deeper understanding of language semantics, making it particularly effective for a wide range of natural language processing tasks (Devlin et al., 2019). Petroni et al. (2019) hypothesize that the BERT model can already possess knowledge about relations based on trained corpora and conclude that it already performs well for the question-answering task, and specific factual knowledge is learned better compared to other language models.

The ability of the BERT model to achieve state-of-the-art performance has led to its extensive use across various applications, though when the model is used on domain-specific data with language-related specifics such as social media, the model's performance is not stable (Caselli et al., 2021). Caselli et al. (2021) re-trained the English BERT base-uncased model and named it HateBERT, the model was re-trained on the publicly available large-scale dataset consisting of English Reddit comments from communities removed for violating platform guidelines related to the offensive, abusive, or hateful nature of their comments. The data set contained 1,478,348 messages (for a total of 43,379,350 tokens), the retraining process took 100 epochs with around 2 million steps in batches of 64 samples, including up to 512 sentencepiece tokens, the learning rate was set to 5e-5 (Caselli et al., 2021). As a

result, HateBERT is a model capable of identifying offensive, abusive, and hateful language.

Since this model was specifically fine-tuned in hateful comments for the task of hate speech detection, we evaluate the performance of this model on our dataset to compare with BERT and select the best performing (*see Section* 5.1).

2.5 Roberta model for hate speech detection

Another masked language model that we used for our experiments is RoBERTa (Robustly Optimized BERT Pre-training Approach). This model is based on the BERT (Devlin et al., 2019) model's architecture, but it was trained differently: the model was trained for a longer amount of time, with larger batches, with more data, and without the next sentence prediction objective (Hugging Face, 2025).

RoBERTa model is often used for various NLP tasks, in particular for hate speech recognition (Alonso et al., 2020; Xu et al., 2020), therefore, we will use this model for the comparison with the BERT model for our dataset with three major classes (*explicit hate*, *implicit hate*, *not hate*) to have an overall understanding of the performance of the base models.

2.6 Generative models for classification task

2.6.1 Qwen2-7B Instruct model

Large Language Models represent the latest stage in the development of models used for NLP tasks: from supervised machine learning, to neural networks with engineered architectures, and finally to LLMs that have fixed architectures and are pre-trained on large textual datasets (Liu et al., 2023). While traditional supervised learning trains models to predict an output given an input, LLMs are prompted (instructed with text) to directly predict the probability of observed textual data (Liu et al., 2023). By carefully crafting prompts (a process known as "prompt engineering"), one can influence the behavior of the model to produce the desired result (Liu et al., 2023).

The extensive pre-training of LLMs on large textual corpora enables their broad applicability in various NLP tasks. For example, GPT-3, a Generative Pre-trained Transformer language model with 175 billion parameters, was trained on 300 billion tokens, and when it was applied without any gradient updates or fine-tuning, only by textual prompting, it achieved strong performance on many NLP datasets, including translation, question-answering, and cloze tasks, etc. (Brown et al., 2020).

Qwen2 series models are transformer-based large language models developed by Alibaba (QwenTeam, 2025). The model architecture was built on the Meta AI LLaMA architecture, and the model was trained using the prediction of the next token (QwenTeam, 2025). The models were pre-trained on large corpora of more than 7 trillion tokens, covering various domains and languages (Yang et al., 2024). Compared to previous versions, Qwen2 models were pre-trained in a more diverse and extensive textual corpus, which was expected to enhance their reasoning capabilities (Yang et al., 2024). The Qwen2 series includes a number of base language models and instruction-tuned language models ranging from 0.5 to 72 billion parameters. During the post-training phase, supervised fine-tuning and direct preference optimization were applied to align model behavior with human preferences by learning from human feedback, thus enabling the models to demonstrate effective enhanced instruction follow-up capabilities (Yang et al., 2024).

The weights of the Qwen series models are open to the public; this open-source nature of the models makes it accessible for research, enabling further fine-tuning and experimentation. At the moment of its release in June 2024, the models generally outperformed most open source models and showed strong competitiveness against proprietary models on a wide

range of benchmarks in language understanding, generation, multilingual ability, coding, mathematics, and reasoning compared to state-of-the-art open source language models and the Qwen1.5 model (Yang et al., 2024). The evaluation of the Qwen series models showed that the Qwen2-72B model outperformed Llama-3-70B in general knowledge understanding on both MMLU and MMLU-Pro, achieving accuracy improvements of 4.7 and 2.8, respectively; Qwen272B had reasoning capabilities that were on par with Llama-3-70B (Yang et al., 2024).

Performance results of the Qwen2-7B model with 7 billion parameters compared to previously released state-of-the-art 7B+ models including Mixtral-7B, Gemma-7B, Llama-3-8B and Qwen1.5-7B demonstrated superior performance of the Qwen2-7B model in most benchmark datasets, especially outperforming other models in coding tasks and mathematics, and demonstrated strong performance in multilingual understanding (Yang et al., 2024). Comparison of Qwen2-7B-Instruct with recent SOTA models with 7-9 billion parameters, including Llama-3-8B-Instruct, Yi-1.5-9B-Chat, GLM-4-9B-Chat, and Qwen1.5-7B-Chat, demonstrated that Qwen2-7BInstruct achieves competitive performance against Llama-3-8B-Instruct (Yang et al., 2024). These results suggest that Qwen2-7B was effectively optimized for a broad spectrum of language and reasoning tasks, highlighting its versatility and advanced generalizability (Yang et al., 2024).

2.6.2 Prompting

With the rapid development of generative models, the interest in their capabilities in solving various NLP tasks has increased. More and more research is focusing on prompting methods (Han and Tang, 2022; Guo et al., 2023). Prompt engineering can be viewed as a form of masked language modeling that operates without the need for task-specific fine-tuning (Gao et al., 2021). The main areas on which researchers focus include the prompt structure, the response format, the number of training examples, and the organization of training examples. We experimented with the generative model in our study by implementing different prompt structures and settings.

Zero-shot, one-shot, and few-shot setting

A generative model prompting can be performed in a zero-shot setting when no examples are provided to the model, but only the task description; one-shot setting, when the model is given one example, and a few-shot setting when the model is fed a few training examples.

Brown et al. (2020) concluded that the performance of the GPT-3 model improves when a task description and the number of examples that serve as a context increase. They explained that in the few-shot setting, the model received the task description and training examples with a desirable result per each example; in the few-shot setting, the model's performance reached the state-of-the-art performance (*see Table* 2.1) (Brown et al., 2020). Zhao et al. (2021) agreed that with few-shot "in-context" learning, it was possible to achieve state-of-the-art performance without the need to fine-tune the model. The advantages of the few-shot setting lie in reproducibility: there are no additional computational requirements because the same model is used for the same task (Zhao et al., 2021).

The conclusions mentioned above are similar to the research of hate speech recognition using the generative model conducted by Han and Tang (2022). Prompting experiments were implemented in a zero-shot and a few-shot setting with various numbers of training examples, which is called "in-context learning". Even in a zero-shot setting with thoroughly formatted instructions, the model (GPT-3) already achieved a relatively high level of accuracy (Han and Tang, 2022). They concluded that descriptive instructions and the number of training examples impacted performance the most; and that only after the number of training examples reached eight per class, the model was able to improve performance, which did not further increase with the higher number of training examples (Han and Tang, 2022). The findings of this research formed the foundation for one of our experiments with prompting in different settings.

Task description:
Translate English to French:
Examples:
sea otter = loutre de mer
peppermint = menthe poivrée
plush girafe = girafe peluche
cheese =

Table 2.1: Few-shot setting (Brown et al., 2020)

Prompt structure

Guo et al. (2023) conducted research on the Chat-GPT performance for hate speech detection using various prompting methods (*see Table* 2.2). As a result of the experiments, ChatGPT consistently outperformed the BERT and RoBERTa models on all data sets used for the experiments, with the highest recall (0.97) in the Few-shot setting (Guo et al., 2023). We will explore the findings of this research in our prompt-based experiments by incorporating various components described in the study: definition of hate speech, training examples.

Method	Description
General Prompt (GP)	the model is asked if a sentence is hate speech
General Prompt with Hate Speech Definition (GPwDef)	the model is asked the same general question but is also provided an explanation on what a hate speech is
Few-Shot Learning Prompt (Few-Shot)	the model is provided with examples along with the multiple choice answer Yes/No and prompted to answer a question

Table 2.2: Hate speech detection with various prompting methods (Guo et al., 2023)

Zhao et al. (2021) suggested considering three factors for an efficient prompt design:
1. Prompt format, that can be a template with placeholders, e.g., "Post": post, "Label": label.
2. Training examples that are given to the model in various batches: zero-shot, one-shot, few-shot.
3. The order refers to the sequence in which the training examples are presented to the model. This factor is considered impactful, as neural models typically update their hidden layers in a sequential manner, processing input tokens from left to right.

Zhao et al. (2021) concluded that the arrangement of training examples was more important to the performance of the GPT series model than the number of training examples. Generative models can be biased towards certain answers (e.g., in evaluation of sentiment the model more often provided positive answers), a model's bias toward certain answers can be estimated by providing a content-free input, such as the sample that does not contain any meaningful information (Zhao et al., 2021):
Input: Subpar acting. Sentiment: Negative
Input: Beautiful film. Sentiment: Positive
Input: NA, Sentiment: .

The answer format, such as the order in which the answer options are listed and replacement of the answer choice symbols with uncommon symbols (e.g., instead of "A,B,C,D") impacted the performance of the model. Changing the placement of the right answer choice led to a lower performance of the Llama series model (Alzahrani et al., 2024).

Chapter 3

Data Description

3.1 Data collection

Data from research conducted by ElSherief et al. (2021) were used as training data for masked language models and test data for masked language models and generative model. The data set consists of Twitter posts that were exchanged between members of the most prominent U.S. hate groups (*see table* 3.1) (ElSherief et al., 2021).

U.S. hate groups	**Proportion in data**
Black Separatist	27.1%
White Nationalist	16.4%
NeoNazi	6.2%
Anti-Muslim	8.9%
Racist Skinhead	5.1%
Ku Klux Klan	5.0%
Anti-LGBT	7.4%
Anti-Immigrant	2.12%

Table 3.1: U.S. hate groups representation in dataset by ElSherief et al. (2021)

Data collection was implemented by selecting the three accounts of hate groups per ideological group that had the highest number of followers (*see table* 3.1) (ElSherief et al., 2021). In total 4,748,226 tweets were selected from these hate groups between January 1, 2015 and December 31, 2017 (ElSherief et al., 2021). The researchers performed the part of speech tagging to find the most representative sample, then identified the top 25 terms associated with each ideology; an additional screening of tweets classified as neutral or hateful was conducted, and any tweets that contained explicit keywords found in NoSwear (Jones, 2020) or Hatebase (Hatebase, 2025) were excluded (ElSherief et al., 2021). The data set focuses mainly on implicit hate posts. To achieve this goal, not only explicitly hateful but also offensive posts were removed (ElSherief et al., 2021) "by running a 3-way HateSoner classifier", in the final dataset the explicit hate posts count is only 5% (*see table* 3.2).

3.2 Annotation process

The annotations for the higher-level (implicit class) labels (*see Table* 3.2) were conducted by Amazon Mechanical Turk annotators, who had examples of posts for each label: *explicit hate*, *implicit hate*, and *not hate*. Each post was annotated by three annotators, 95.3% of the posts reached majority agreement (ElSherief et al., 2021).

The annotation of finer-grained labels (i.e., implicit class labels) required a more nuanced and context-sensitive understanding. Three research assistants, who served as expert annotators, were responsible for labeling the data with fine-grained implicit class labels (ElSherief et al., 2021). Throughout the annotating process, the researchers measured annotator agreement Fleiss' Kappa remained at the level of 0.55-0.61 (ElSherief et al., 2021).

3.3 Categories

On a higher level, the data was annotated with three labels (*explicit hate*, *implicit hate*, *not hate*) (*see Table* 3.2). 13,291 posts are labeled as *not hate*, 7,100 posts as *implicit hate* and 1,089 as *explicit hate*.

Higher level labels	**Class**	**Count**	**Proportion in data**
	Not hate	13291	62%
	Implicit hate	7100	33%
	Explicit hate	1089	5%
	Total	**21480**	
Finer level labels (implicit class)	**Implicit class topics**	**Count**	**Proportion in data**
	White Grievance	1538	24%
	Incitement	1269	20%
	Stereotypical	1133	18%
	Inferiority	863	13%
	Irony	797	12%
	Threatening	666	10%
	Other	80	1%
	Total	**6346**	

Note: There is data discrepancy between major labels and implicit class labels for implicit hate: 754 sentences were not labeled with implicit class labels.

Table 3.2: Various labels count with proportion

At a finer level, the same data were annotated with seven implicit class labels describing the topic of a statement (*White Grievance*, *Incitement*, *Stereotypical*, *Inferiority*, *Irony*, *Threatening*, *Other*) (*see Table* 3.2). The label *Other* classified tweets for "any other situations." The labels of the implicit class are not mutually exclusive, but capture the main topics of implicit hate (ElSherief et al., 2021). Although they may not account for every possible case, they cover 98.6% of implicit hate in a sample of the most prevalent hate ideologies in the U.S., such as "black separatist", "white nationalist", etc. (*see Table* 3.1). The seven implicit class labels outlined below represent different thematic categories.

White Grievance posts express the white supremacist discourse with the main characteristics such as victimization, defense of "white" privileges, threatened survival of "whites", etc. (*see data examples in Table* 3.3) (Berbrier, 2000).

Incitement uses rhetoric of fear, threats, dehumanization of opponents, etc. It publicly urges violent action against opponents (*see data examples in Table* 3.3) (Somerville, 2011).

Inferiority language frames one group or individual as lesser than another, often through expressions of dehumanization, assertions of superiority and dominance, attacks on dignity, and portrayals of the targeted group as toxic or dangerous (toxification) (ElSherief et al., 2021). Toxification refers to the cognitive framing of a target group as inherently threatening

to the survival of the perpetrators and the stability of their society; unlike dehumanization, which views the group as subhuman or inferior, toxification casts them as a dangerous contaminant that must be eradicated to protect the society (*see data examples in Table* 3.3) (Neilsen, 2015).

Higher level labels (major classes)	
Post	**Class**
Not hate	has a problem with ” the whites ” ” and ” ” the christians . ” ” go ahead you are free to say it now ! ”
Implicit hate	” : jewish harvard professor noel ignatiev wants to abolish the white race via #wr ” ” ”
Explicit hate	rt crooked hillary you didn't lose because you couldn't portray yourself as an ” agent of change ” ” you loss because you are a socialist agent of destruction . shut up hillary and stop whining ! ”
Finer level labels (implicit classes)	
Post	**Implicit class**
White Grievance	” : jewish harvard professor noel ignatiev wants to abolish the white race via #wr ” ” ”
Incitement	i'm not a racist at all but i mean all the numbers are saying caucaisian people are set to disappear soon so iḿ for keeping the race
Stereotypical	hunt for white women : black racist serial killer confesses to 41 murders |read more : via
Inferiority	:and throw in weekly standard these filthy white trash disgust me
Irony	rt : tune in today's jmt for my interview w / robert spencer on ” the complete infidel's guide to iran ! ” ” ”
Threatening	they smother the good white people of america ! they need to be hung brother ” south ” ” ! ”
Other	right ? what good is ” white supremacy ” ” if it leads to a white minority . ”

Note: This table contains unrevised data examples with all grammar and spelling kept as provided in the data set by ElSherief et al. (2021).

Table 3.3: Data examples

The **Irony** uses sarcasm in forms of hyperbole, understatement, and mocking language

and serves to communicate insult (*see data examples in Table* 3.3) (Justo et al., 2014).

Stereotypical trait frequently relies on familiar stereotypes to demean or belittle an individual or group; and because stereotypes are often embedded in context, language can be perceived as hateful even when no individual word is explicitly offensive (Warner and Hirschberg, 2012). Misinformation is often associated with stereotypes as the tool of the latter (*see data examples in Table* 3.3) (ElSherief et al., 2021).

Threatening may not express direct incitement, it still attributes negative traits or qualities to a targeted group; it resembles insults or stereotype-based judgments and may imply that perceived negative traits pose a threat (*see data examples in Table* 3.3) (Sanguinetti et al., 2018).

3.4 Lexical analysis

For further analysis and to gain deeper insight into the data, we conducted a lexical analysis by extracting the most likely tokens for selected categories by using log-likelihood testing. Log-likelihood evaluation is a statistical tool that checks how statistically significant the difference in word usage is between categories (Dunning, 1993). We identified the most distinctive tokens for each category of each implicit class (*see Table* 3.4).

Top *Incitement* tokens	**Top *Inferiority* tokens**	**Top *Irony* tokens**	**Top *Stereotypical* tokens**	**Top *Threatening* tokens**	**Top *White Grievance* tokens**	**Top *Other* tokens**
aryan	iq	difference	kill	deport	anti	cuck
hitler	animal	number	jewish	send	genocide	cuckservative
resister	low	repeat	rape	ice	racist	asexual
adolf	civilization	ethiopian	jihad	em	whitegenocide	adam
alt	rat	jerome	israel	gun	hate	someday
shirt	savage	car	old	dank	code	slogan
kkk	parasite	joke	koran	round	antiwhite	skrillex
power	monkey	cotton	muslims	nuke	accelerate	sikh
pride	degenerate	parrot	infidel	ship	privilege	settle
whitepride	pig	tree	islamic	period	minority	rude

Table 3.4: Implicit class tokens by likelihood

For data analysis, we checked the most frequent tokens for every class. We list the most frequent tokens by class from most frequent to least frequent in the table (*see Table* 3.4).

The most frequent tokens for the **White Grievance** class are: "whitegenocide", "anti", "genocide", "racist", "antiwhite", "privilege", etc. (*see Table* 3.4).

The most frequent tokens for the **Incitement** ("aryan", "hitler", "resister", "adolf", "alt", "kkk", "power", "whitepride", etc.) are strongly associated with alt-right propaganda narratives (*see Table* 3.4).

The **Inferiority** class operates with tokens like "iq", "animal", "low", "rat", "savage", "parasite", "monkey", "degenerate", etc. (*see Table* 3.4).

The **Irony** expression is subtle, therefore, the most frequent tokens do not categorize the speech clearly as hateful, for example, token "difference" is followed with "number", "repeat", "ethiopian", etc. (*see Table* 3.4).

The most frequent tokens in the **Stereotypical** class call to "kill", which is the most frequent token, along with such tokens as "jewish", "jihad", "israel", "koran", "muslims", etc. (*see Table* 3.4).

The most frequent tokens for the **Threatening** class unlike other classes contain verbs: "deport", "send", "gun", "ship" which along with the noun "ice" (i.e., "U.S. Immigration and Customs Enforcement"), and the pronoun "em" already identify the discourse of this class (*see Table* 3.4).

The most frequent tokens for the **Other** class include words like "cuck", "cuckservative", "asexual" (*see Table* 3.4).

Chapter 4

Methodology

We conducted experiments with masked language models to establish a baseline. And we carried out experiments with the generative model to identify which prompting method described by Han and Tang (2022) and Guo et al. (2023) achieves the best performance with a specific focus on implicit hate speech.

4.1 Experiments with masked language models (BERT, HateBERT, RoBERTa)

Experiment	**Total dataset**	**Train**	**Dev**	**Test**
Three labels: BERT-base-uncased	21480	12888	4296	4296
Three labels: HateBERT	21480	12888	4296	4296
Three labels: RoBERTa	21480	12888	4296	4296
Finer labels: BERT-base-uncased	20726	12474	4126	4126
All prompts for Qwen				4126

Note: The first three experiments were conducted on the data labeled with major labels. The "Finer labels: BERT-base-uncased" and "All Prompts for Qwen" were conducted on the same data labeled with implicit class labels. There is a data discrepancy between major labels and implicit class labels for implicit hate: 754 messages were not labeled with implicit class labels.

Table 4.1: Experiments and data

Three masked language models (BERT-base-uncased, HateBERT, RoBERTa) were fine-tuned and tested. We ran these three models to evaluate how our data performed with different models. We checked the BERT-base-uncased (Devlin et al., 2019) because it is often referred to as a baseline for various NLP tasks (Petroni et al., 2019). Caselli et al. (2021) observed that HateBERT, which was specifically re-trained for the hate speech detection task, outperforms the BERT model. We ran HateBERT to check if it outperformed the BERT-base-uncased. And we checked the RoBERTa model because, based on various research (Alonso et al., 2020; Xu et al., 2020; Liu et al., 2019), it consistently achieved higher than BERT results for various NLP tasks. We compared the performance of these three models. The best-performing model was identified as the baseline that we referred to when evaluating the performance of the generative model.

Implicit Class	Sentence Count
Not hate	2658
Explicit hate	218
White Grievance	301
Incitement	248
Stereotypical	221
Inferiority	172
Irony	159
Threatening	133
Other	16
Total	4126

Table 4.2: Test set labels distribution

First, we used data labeled with major (higher-level) labels: *explicit hate*, *implicit hate*, *not hate*. The test set used for BERT was also used for Qwen. There is a data discrepancy of 754 messages between higher-level labels and finer-level labels (*see Table* 4.1). We kept only messages with finer-level labels. We followed the data split described by ElSherief et al. (2021): the data was split into a training data set (60%), a development data set (20%), and a test set (20%) (*see Table* 4.1).

Second, we used the same data labeled with finer *implicit hate* labels. The goal of this experiment was to check whether the masked language model performed better if it saw the data with finer labels. We trained and tested the best-performing masked language model, i.e., BERT-base-uncased. BERT-base-uncased was fine-tuned with nine classes, and classification was done back to three classes. The model was fine-tuned with the following nine classes: *not hate*, *explicit hate*; and seven implicit hate classes: *White Grievance*, *Incitement*, *Stereotypical*, *Inferiority*, *Irony*, *Threatening*, *Other*. The results were reclassified into the following three major classes: *not hate*, *implicit hate*, *explicit hate*.

We kept all hyperparameters for the masked language models the same for all experiments. Following the recommendations by Devlin et al. (2019), we set the hyperparameters at the following levels. We kept the batch size = 16 due to computing constraints. Three learning rates are recommended by Devlin et al. (2019): 5e-5, 3e-5, 2e-5. We selected the learning rate = 5e-5, because although the three hyperparameters are not high, 5e-5 would allow us to achieve faster training. The number of epochs as 2,3,4, is recommended by Devlin et al. (2019); we set the number of epochs = 2 due to computing constraints and a relatively small dataset.

4.2 Experiments with generative model (Qwen)

After establishing the baseline, we experimented with the generative model. The generative model (Qwen) was used for a prompt evaluation. We selected this model because it is open source, which means we were able to investigate not only prompting methods, but also parameters and their impact on the model's performance.

We aimed to investigate one of the research questions: the performance of various prompting methods for the detection of hate speech by the generative model. Before checking various prompting techniques, we conducted brief experiments with prompt engineering:

1. We briefly checked the generative model's parameters tuning to understand how switching the model's parameters from one of the highest to the lowest and then average impacted the clarity and precision of the generated answer.

2. We briefly checked the prompt format to understand how the structure of the prompt impacted the model's performance.

3. We identified which "role" was the most suitable for the goal of our study.

4. We briefly checked if phrasing a prompt as a question or instruction affected the clarity of the generated answer.

4.3 Generative model parameters

This section describes a brief experiment with different parameters. The experiment was carried out solely to check if variations in parameters influenced the accuracy of the generated responses and to select the set of parameters suitable for application on the test data set of 4126 messages. The experiment is non-systematic and does not cover the whole development data set. As test data samples, we selected the most representative data samples of *not hate*, *implicit hate*, *explicit hate* from the development dataset. By "the most representative data samples", we mean the instances that clearly exemplify the defining characteristics of each label. For example, for the testing example labeled as *not hate*, we ensured the absence of slurs or overtly violent language that could be easily interpreted as hateful (e.g., "People in real trouble don't have the luxury of being antisocial."). For the testing example labeled as *explicit hate*, we selected an example containing direct slurs or offensive expressions (e.g., "Trump is a white trash."). Finally, for the test example labeled as *implicit hate*, we included an example featuring coded or indirect language, like sarcasm and metaphor (e.g., "Just cut aid and let the Africans govern themselves into the ground.").

We checked the parameters using the general prompt with the definition of hate speech, which was inspired by the research of Guo et al. (2023). For example, the prompt for the test sample labeled as *explicit hate* had the following format: "Do you think this comment is *not hate*, *explicit hate*, or *implicit hate* speech? Explicit hate speech is any communication that disparages a person or a group on the basis of some characteristic such as race, color, ethnicity, gender, sexual orientation, nationality, religion, or other characteristics. Implicit hate speech is a coded or indirect language, such as sarcasm, metaphor, circumlocution, etc., used to disparage a protected group or individual, or to convey prejudicial and harmful views about them. Test sample: "Trump is a white trash."

Liu et al. (2023) considered various methods to adjust the parameters that affect a language model or prompts. A key design choice involved determining which parameters to update during training, as this decision significantly influenced the adaptability of the method in different application scenarios (Liu et al., 2023). The authors argued that prompting could alleviate the issue with limited training data and help a model predict the probability (Liu et al., 2023).

We conducted a brief experiment by setting certain parameters to different levels and analyzing the responses generated. We selected the following parameters:
- temperature = 0.0, 0.5, 1.0,
- top_p = 0.0, 0.5, 1.0,
- presence_penalty = 0.0, -2.0, 2.0,
- frequency_penalty = 0.0, -2.0, 2.0,
- max_tokens= 256, 5.

Temperature is used to predict the next token probability (Hugging Face, 2024) and controls the randomness of the generated output; higher temperature increases variability, while a lower temperature results in a more deterministic and predictable output (Hugging Face, 2024). At a temperature = 1.0, the generated response was vague and described the harmful potential of the post. At 0.0, the answer was more focused on the message and

produced the label. At 0.5, the answer was somewhat less vague compared to 1.0, but still provided the label and commented on in somewhat unclear general terms.

Top_p parameter picks from the most likely tokens whose total probability reaches the set threshold. If it is set to a float less than 1, only the smallest group of the most likely tokens (whose combined probabilities meet or exceed top_p) are kept (Hugging Face, 2024). At top_p = 0.0, only the words with the highest probability are selected, and at top_p = 1.0, the whole set of tokens is considered (MetaAI, 2024). We observed that at top_p = 0.0, the generated answer was clear and focused. At top_p = 1.0 the answer was more elaborate and vague. At top_p = 0.5 the model did not provide a definitive answer, was more ambiguous, suggesting that the message could be both *explicit* and *implicit hate* speech.

The **max_tokens** sets the total maximum number of tokens in the generated sequence, including the input prompt (Hugging Face, 2024). Any output larger than specified max_tokens is cut off (Cloud, 2025). The Qwen2-7B-Instruct model can generate a maximum of 4096 tokens (NVIDIA, 2025). We initially set this parameter to 256 to restrict the model from producing overly verbose responses while assessing the accuracy of its outputs. Upon observing that this parameter influenced the length of the generated comments but had no effect on their accuracy, we subsequently reduced its value to 5.

The **frequency_penalty** penalizes the model if it repeats the same tokens, with higher values reducing repetition and lower values allowing repetition (de la Vega, 2023). At frequency_penalty = -2.0, the model failed to generate a logical answer that also contained repetitive words. At frequency_penalty = 2.0, the answer was extremely vague and identified a post as both *implicit hate* and *explicit hate*. At 0.0, the model labeled the message with only one label and justified the answer.

The **presence_penalty** reduces the probability in the model reusing the same tokens for text generation, a higher value discourages reusing of the same tokens, making the text more creative, a lower value allows reusing tokens (de la Vega, 2023). The change in settings did not affect the response style or length; for all three settings, the model clearly labeled the message with the same label.

Based on the results, we conclude that some parameters affect the precision of the answer (for example, the model labeled a message as both *explicit hate* and *implicit hate*, when the top_p was set = 0.5), but for most parameters the change in the setup to three different levels did not cause the model to change the generated answer. What the parameter switch affected was the clarity of the generated answer; therefore, for all prompting experiments, we set parameters (temperature, top_p, frequency_penalty, presence_penalty) to = 0.0.

4.4 Prompt format

Initial structure:
Do you think this comment is *not hate*, *explicit hate* or *implicit hate* speech? Explicit hate speech is any communication that disparages a person or a group on the basis of some characteristic such as race, color, ethnicity, gender, sexual orientation, nationality, religion, or other characteristics. Implicit hate speech is a coded or indirect language, such as sarcasm, metaphor, and circumlocution, used to disparage a protected group or individual, or to convey prejudicial and harmful views about them.
Comment: Trump is white trash.

Final structure:
Explicit hate speech is any communication that disparages a person or a group on the basis of characteristics such as race, color, ethnicity, gender, sexual orientation, nationality, religion, or other characteristics. Implicit hate speech is a coded or indirect language such as sarcasm, metaphor, circumlocution, etc. used to disparage a protected group or individual, or to convey prejudicial and harmful views about them. Classify the following comment by responding with ONLY ONE of these exact labels and nothing else:
not_hate
explicit_hate
implicit_hate
Comment: "{post}"
+ reinforcement message at the end of the prompt:
Respond ONLY with one of the above labels.

Table 4.3: General prompt with hate speech definition

Guo et al. (2023), some of whose prompt strategies we adopted (*see Table* 2.2), framed their prompt as a general question "Do you think. . . ?" To identify whether this structure works for our data, we conducted brief experiments by modifying a prompt structure. When asked "Do you think...?", the model provided a comprehensive answer with reasoning and definitions of hate speech varieties. Since the prompt would process a large dataset, we aimed to get only labels as generated answers. Han and Tang (2022) structured their prompts as instructions "Classify...", and Liu et al. (2023) suggested adding to the prompt explanations about the "desired result" and listing the answer options under bullet points. Such a structure should help the model to achieve clarity of the answer. After rephrasing the question to the instructive one "Classify...", the model started giving shorter answers, but still was trying to answer in complete messages by quoting the test sample's segment it found hateful. Zhao et al. (2021) identified that the generative model tends to repeat tokens that appear at the end of the prompt; we placed a "reinforcement message" at the very end of the prompt. This message emphasized the desired outcome and listed the labels again. We observed an improvement in the generation of answers after restructuring the prompt. The initial and final structures of a prompt are illustrated in the example of a general prompt with definition of hate speech (*see Example* 4.3).

The experiment proved that it is important to formulate the answer in the exact format in which we need the answer to be given. Additionally, the model tends to "forget" the instructions given at the beginning of the prompt; therefore, the same instructions should be repeated at the end of the prompt with reinforcement attention words (e.g., "IMPORTANT") typed in uppercase: "IMPORTANT: Your entire response must be EXACTLY ONE of these three labels with no explanation, no reasoning, and no additional text: *not hate*, *explicit*

hate, *implicit hate*." This formatting helped us avoid undesirable answer formats in longer prompts, such as the Few-shot prompt, which contains eight training examples per label (*see the Prompt* 2).

4.5 "Role" modifications

The Qwen model architecture is similar to the Llama model's architecture, therefore we could experiment with the same roles that are suggested by Llama, in particular "User" (the model will consider the inputs from users and "includes the inputs, commands, and questions to the model") and "Assistant" (the model will generate the answer "based on the context provided in the system, ipython and user prompts") (Hugging Face, 2024). The core difference between "Assistant" and "User" roles lies in their underlying architecture: the "User" is more focused on the provided input and commands, while the "Assistant" considers context provided in the system (Hugging Face, 2024).

Following the research by Chen et al. (2024), we switched the roles from "Assistant" to "Helpful assistant" and then to "User" without changing the prompt question format "Do you think this comment is *not hate*, *explicit hate*, or *implicit hate* speech? Test sample: white community was a safe white community till moslems / islam came, what a shame." The "Assistant" suggested that the comment "does not directly use strong, explicit language" and then defined implicit hate speech. The "Helpful assistant" answered in the format of continuous chat: "Sure, I can help you with that! What do you need assistance with?". The answer of the "User" was clearer and better structured (unlike the answer of the "Assistant" that, in some cases, started mid-sentence); it identified the comment as "implicit hate speech", gave its definition, and implied that implicit hate speech "can vary based on context and legal definitions in different regions."

The model answered wrong in all cases when prompted with an open-ended question "Do you think...", and did not suggest any answer when set up as a "Helpful assistant".

Our experiment aimed to process larger test data samples; therefore, it was important to obtain definitive answers without the necessity of post-processing the data; as such, we set the "role" to the "User" (*see Table* 4.4). We observed that the "User" role provided clearer and more structured answers.

Role	**Prompt**	**Answer**
Assistant	**Do you think this comment is** not hate, explicit hate or implicit hate speech? *Test sample*: white community was a safe white community till moslems / islam came what a shame	"The test result shows that the comment contains implicit hate speech." + one paragraph of justifying comments
Helpful assistant		"Sure, I can help you with that! What do you need assistance with?"
User		*implicit hate* + four paragraphs of detailed comments justifying the reasons why the choice is not an *explicit hate*

Table 4.4: Role modifications

4.6 Prompt strategies

The purpose of our experiments with various prompting methods was to identify among the prompts suggested by Guo et al. (2023) and Han and Tang (2022) the prompt with the best

performance by adapting some features of the suggested prompt strategies. We used the prompt structure of the "in-context learning prompt" by Han and Tang (2022) as a basis of our prompting method: 1) Instructions ("Classify the text...") + 2) Training example and answer 3) Test sample. Training examples were removed if the prompt followed the General prompt structure (*see Table* 4.5). Our first set of experiments used the major (higher-level) labels (*explicit hate*, *implicit hate*, *not hate*) (*see Table* 4.5). For these experiments, we used the same test set that we used for the masked language model experiments (4126 messages) (*see Table* 4.1). The test set contained major (higher-level) labels (*explicit hate*, *implicit hate*, *not hate*) and implicit class (finer-level) labels (*White Grievance*, *Threatening*, *Incitement*, *Inferiority*, *Irony*, *Stereotypical*, *Other*).

To evaluate how various prompting techniques affected the performance of the generative model, we conducted experiments with the major (higher-level) labels. With each subsequent prompt, some additional or new information was added to the prompt. The prompts with finer labels contained implicit class labels; by adding them, we investigated whether adding finer labels improved the generative model's performance.

Major labels prompts

We started with prompting strategies utilizing major labels: General prompt (Exp5-GP), General prompt with the definition of hate speech (Exp6-GP+Def), One-shot prompt (Exp7-GP+1shot), Few-shot prompt (Exp8-GP+FS) (*see Tables* 1, 2).

General prompt (Exp5-GP), General prompt with the definitions of explicit and implicit hate speech (Exp6-GP+Def) prompting strategies were inspired by research conducted by Guo et al. (2023) (*see Table* 1 for Exp5 - GP (General prompt) and Exp 6 - GP + Def (General prompt with hate speech definition)). We selected these prompting methods to check whether additional information about hate speech helped the model improve its performance. The General prompt (Exp5-GP, *see Table* 4.5) is a simple instruction to classify a post with one of the given labels.

Guo et al. (2023) investigated which prompting strategies most effectively used the LLM knowledge base for contextual hate speech detection. The authors selected the General prompt structure developed by Li et al. (2023), who demonstrated the effectiveness of this prompt in the detection of hate speech (Guo et al., 2023). The authors described that the General prompt engineering techniques enabled the adaptation of LLMs to the specific task of hate speech detection. For every message, the model was prompted to provide an answer on the question: "Do you think this comment is hate speech? comment: {x} a. Yes b. No. The model output y, either "a. yes" or "b. no"(Guo et al., 2023). As described in the section "Prompt format" (*see Section* 4.4), we modified the General prompt as instruction prompt "Classify..." to avoid undesirable answer formats.

The General prompt with hate speech definition (Exp 6-GP+Def) contains *implicit hate* and *explicit hate* speech definitions. Guo et al. (2023) conducted their research on various hate speech datasets. To deal with variations in different datasets, they further added to the General prompt the definition of hate speech. They hypothesized that the additional context would help LLMs understand what hate speech was and evaluate LLMs' performance (Guo et al., 2023). Since our experiments were conducted on a dataset containing various implicit hate classes and involved two distinct types of hate speech, we adopted this prompting strategy and included the definitions of *explicit hate* and *implicit hate* into our prompt (*see Table* 1).

In addition, we also used the One-shot prompt (Exp7-GP+1shot) and the Few-shot prompt (Exp8-GP+FS) for major labels (*see Table* 1 for the One-shot prompt (Exp7-GP+1shot) and *see Table* 2 for the Few-shot prompt (Exp8-GP+FS)). These prompts were inspired by the research conducted by Han and Tang (2022), who explored the concept of "in-context learning": a pre-trained LLM was provided with a test example, task description, and a few training examples without modifications in the model parameters. The authors

Masked language models experiments			
Experiment	**Setup**	**Classes**	**Description**
Exp 1 - B	BERT	3 major	(*See "Experiments with masked language models (BERT-base-uncased, HateBERT, RoBERTa)" section in thesis* 4.1)
Exp 2 - Rb	RoBERTa	3 major	Same as BERT
Exp 3 - hB	HateBERT	3 major	Same as BERT
Exp 4 - B - Finer	BERT	2 major + 7 finer	Classification into finer-grained classes
Generative model experiments			
Exp 5 - GP	General Prompt	3 major	No definitions, no examples. Base instruction prompt
Exp 6 - GP+Def	General Prompt + HS definitions	3 major	Definitions of *explicit* and *implicit hate* speech. No examples.
Exp 7 - GP+1shot	General Prompt + One-shot	3 major	One example per class (*explicit hate*, *implicit hate*, *not hate*); no definitions.
Exp 8 - GP+FS	General Prompt + Few-shot	3 major	8 training examples per class, 24 examples in total; no definitions.
Exp 9 - GP + FS + (Consider Finer)	Few-shot prompt + General Prompt with Finer Labels	3 major	8 training examples per class, 24 examples in total; no definitions. Baseline instruction prompt + "For *implicit class* consider" finer-level (implicit class) labels.
Exp 10 - GP + (Consider Finer)	General Prompt with Finer Labels	2 major + 7 finer	7 finer *implicit hate* labels + *not hate*, *explicit hate* → mapped into 3 major. No definitions, no examples. Baseline instruction prompt + "For *implicit class* consider" finer-level (implicit class) labels.
Exp 11 - GP + (Classify Finer)	General Prompt with Finer Labels	2 major + 7 finer	7 finer *implicit hate* labels + *not hate*, *explicit hate* → mapped into 3 major. No definitions, no examples. Baseline instruction prompt

Table 4.5: Experiments and setup: various labels were used for different prompts

motivated their choice of this concept by the impracticality of fine-tuning LLM and the urgency of detecting hate speech due to the rapid growth of hateful comments on social media; they argued that detecting hate speech was further complicated with the variety of definitions of hate speech and the differences between hate speech and offensive language (Han and Tang, 2022). They investigated whether adding more training examples improved the performance of the generative model and noted that the model achieved the best performance when asked with eight training examples per label (Han and Tang, 2022). Therefore, in our study, we conducted a 24-shot prompt (eight training examples per each of the three major classes) to evaluate how the prompt worked for our data and compare it with other prompts. The One-shot prompt (Exp7-GP+1shot) contained one training example per label. We used this prompt to further compare the results with the few-shot setting (Few-shot prompt (Exp8-GP+FS)) and the zero-shot setting (General prompt (Exp5-GP)).

As described in the section "Generative model parameters" (*see Section* 4.3), same as for the parameters experiment, the selection of the test data samples for one-shot and few-shot settings was conducted manually. The test samples were selected from the development data set to best represent the classes of *explicit hate*, *implicit hate*, *not hate*.

Finer labels prompts

Our second set of experiments utilized the finer-level (implicit class) labels (*see Table* 4.5).

We aimed to investigate whether adding finer labels improved the performance of the generative model (*see Table* 4.1). We used the same test set that we used for the BERT model experiment and major labels prompts (4126 messages) (*see Table* 4.1) but labeled with implicit hate classes (*White Grievance*, *Threatening*, *Incitement*, *Inferiority*, *Irony*, *Stereotypical*, *Other*) + *not hate*, *explicit hate*.

The prompts for finer labels were inspired by the research of Han and Tang (2022). In addition to exploring the impact of training examples, the authors investigated alternative strategies to enhance precision: messages in the data set contained additional labels that marked the kind of offensive content (violence, gender, race, disability, religion or sexual orientation); these additional labels represented prior knowledge relevant to determining whether a comment constituted hate speech (Han and Tang, 2022). Therefore, they investigated how to incorporate this prior knowledge into the in-context learning framework: first, the model was prompted to classify if a text belonged to one of the offensive labels or not (*"Classify the following texts into 'gender offensive', 'race offensive', 'national origin offensive', 'disability offensive', 'religion offensive', 'sexual orientation offensive' or 'not'"*.); second, the model was instructed to classify over major binary labels (hateful or not) and consider finer labels (*"Classify the following texts into 'hate speech', or 'not'. For 'hate speech', consider if it is gender offensive, or race offensive, or national origin offensive, or disability offensive, or religion offensive, or sexual orientation offensive"*.) (Han and Tang, 2022). The finer-level offensive labels improved the generative model's performance compared to binary classification. The results showed that the model achieved the highest precision (0.87) and recall (0.94) when prompted with the second prompt (Han and Tang, 2022). The first (classification over finer labels and *not hate*) prompt achieved 0.77 precision and 0.92 recall (Han and Tang, 2022). Thus, the model performed better when prompted to classify over binary labels, but consider finer labels rather than classifying over all finer labels and *not hate*. In our study, we used these prompt strategies with some modifications. As an adaptation of the first prompt, we instructed the model to classify across nine labels: two major and seven finer labels in the prompt Exp11-GP+(Classify Finer) (*see Table* 5), the results were mapped back to three major labels. And as an adaptation of the second prompt, we asked the model to classify over three major labels and consider finer labels for *implicit hate* in the prompt Exp10-GP+(Consider Finer) (*see Table* 4) and with some modifications in the prompt Exp9-GP+FS+(Consider Finer) (*see Table* 3). The prompt

Exp9-GP+FS+(Consider Finer) instructed to classify across three major labels and for *implicit hate* consider only six implicit class labels (*see Tables* 3). We excluded the label *Other* because this label, unlike other implicit class labels, does not have any meaning. Therefore, we assumed that the model would not be able to correctly predict any message with the label *Other*. In this prompt (Exp9-GP+FS+(Consider Finer)), we combined the best-performing prompt for the major classes (Few-shot prompt (Exp8-GP+FS) with the finer-level labels to evaluate whether the experiment described by Han and Tang (2022) could be applicable to our data.

The prompts with three major classes (*implicit hate*, *explicit hate*, *not hate*) are in tables 1, 2.

The prompts with finer labels + *not hate*, *explicit hate* are in the tables 3, 4, 5.

4.7 Evaluation metrics

We use accuracy, precision, recall, and the F1 score for quantitative evaluation. Since our data are imbalanced (there are more than twice as many of the *not hate* data points than *implicit hate* data points, and significantly fewer *explicit hate* data samples (*see Table* 3.2), in our performance evaluation we will use macro-average precision, recall, and F-1 score. Macro-average counts are calculated by summing up the results of all classes and then averaging over all classes (Jurafsky and Martin, 2024); this method allows us to have a better representation of the results of minority classes.

Chapter 5

Results and Analysis

5.1 BERT, Roberta models results

The best performing BERT model is BERT base-uncased with the highest macro-average precision, recall, and F1 score (*see Table* 5.1). The BERT base-uncased outperformed a specifically trained for hate speech detection HateBERT model (*see Table* 5.1). The macro-average F-1 for HateBERT is 63.28%. The BERT-base uncased F-1 score is 63.91%. The performance of the BERT and HateBERT models is largely consistent, HateBERT being slightly behind the BERT model.

RoBERTa model's precision is on par with the BERT, but there is a drop in recall for RoBERTa, especially for the *explicit hate* class. With a closer look at the performance per main hate speech category, we observe that HateBERT dealt better with the recognition of *explicit hate* speech with the highest precision, and RoBERTa achieved the highest precision for *implicit hate* speech and recall for *not hate*.

Some research results indicated that the detection of *implicit hate* remained a challenge, while the detection of *explicit hate* produced more consistent and stable results (Caselli et al., 2020). We hypothesized that *explicit hate* would be the class with the best performance. For the *explicit hate* and *not hate* classes, the overall performance correlates with the data count: the *not hate* class has twice as many data samples as the *implicit hate* class, consequently the model has more training samples to learn from, which may be one of the reasons that the class achieves the best performance for all three models. Data filtering was implemented for the *explicit hate* class during the data collection step (*see Section* 3.1). In addition, this class has limited representation in the dataset, which could have impacted class performance. Contrary to expectations, specifically trained for hate speech recognition, HateBERT did not perform better than the BERT base model.

Model	**BERTbase-uncased**			**HateBERT**			**RoBERTa**		
	P	R	F1	P	R	F1	P	R	F1
not hate	83.62	85.05	84.33	82.97	83.53	83.25	81.80	86.05	83.87
implicit hate	66.20	67.14	66.67	64.34	66.86	65.57	66.25	64.22	65.22
explicit hate	51.16	33.85	40.74	53.28	33.33	41.01	50.00	25.13	33.45
Macro-average	**66.99**	**62.01**	**63.91**	66.86	61.24	63.28	66.02	58.47	60.85

Table 5.1: Masked language models' performance by hate speech type and macro-average per metric

ElSherief et al. (2021), whose data set is used in this study, report binary classification results (*implicit hate* vs. *not hate*) for the BERT model, achieving a precision of 72.1%,

recall 66.0%, and F1 score 68.9%. There are no existing studies that have employed the same data set and the BERT model for classification of this data set.

BERT-base-uncased model experiment with finer labels

We conducted the experiment by fine-tuning the BERT-base-uncased model with finer labels and mapping them back to three major classes. The purpose of the experiment was to test whether finer labels could improve the performance of the model compared to the experiment with major labels. The experiment showed that the finer labels BERT do not outperform generic BERT. We observe macro precision (66.99%) of the generic BERT vs. 65.43% of the finer labels BERT, while the recall is slightly higher for the finer labels BERT (*see Table* 5.2). The effect of finer implicit labels is minimal; the BERT with finer labels identified more true positives compared to the BERT trained with only three major labels.

Model	**BERT (3 major labels)**			**BERT (finer labels)**		
	P	R	F1	P	R	F1
not hate	83.62	85.05	84.33	83.83	88.94	86.31
implicit hate	66.20	67.14	66.67	70.00	62.72	66.16
explicit hate	51.16	33.85	40.74	42.47	36.24	39.11
Macro-average	**66.99**	62.01	**63.91**	65.43	**62.63**	63.86

Table 5.2: BERT-base-uncased models performance: BERT with three major labels vs. BERT with finer labels

In our experiments with the masked language models, we used random seeds. This can lead to the potential variability in model performance that arises when running the same experiment multiple times with different random seeds. This variance in task performance arises even when everything else, such as dataset, architecture, hyperparameters, remains constant (Mosbach et al., 2021). Devlin et al. (2019) documented such unstable performance when fine-tuning BERT and suggested experimenting with various seeds on the development data set and selecting the best performing model.

5.2 Qwen model results

We conducted various prompting experiments with major labels and the combination of major labels with finer labels. The methodology for the major label prompts involved incorporating various elements into the base prompt instructions: definitions of *explicit hate* and *implicit hate*, one training example for each of the three major labels (*explicit hate*, *implicit hate*, *not hate*), or a few training examples (eight training examples for each of the three major labels). The methodology for the prompts with finer labels involved modifications of the prompt instructions from classification to a more complex one: classification with consideration of finer implicit class labels (*see Section* 4.6). As no previous experimental results on this data set and with similar prompting techniques are available from other researchers, our findings cannot be directly compared to previous studies.

Based on the performance of each prompt, we can draw some main conclusions.

BERT vs. QWEN

First, the comparison of the results of the best-performing prompt (Exp9-GP+FS+(Consider Finer) with our baseline (BERT-base-uncased) shows that the best-performing prompt achieved lower than the BERT model scores: F1 score (46.07% vs. 63.91%), precision (46.71% vs. 66.99%), and recall (56.12% vs. 62.01%) (*see Table* 5.2, *see Table* 5.3).

Zero-shot vs. one-shot vs. few-shot

Model	Exp5-GP			Exp6-GP+ Def			Exp7-GP+ 1shot			Exp8-GP+ FS		
	P	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
not hate	92.26	26.90	41.65	94.89	23.74	37.98	89.28	42.29	57.39	81.49	66.25	73.09
implicit hate	20.83	14.40	17.03	23.41	22.64	23.02	26.00	17.20	20.70	37.13	26.88	31.18
explicit hate	8.64	98.62	15.90	8.88	91.74	16.19	10.10	94.50	18.25	15.66	76.15	25.98
Macro-average	40.58	46.64	24.86	42.39	46.04	25.73	41.79	51.33	32.11	44.76	56.43	43.42

Model	Exp9-GP+ FS+ (Consider Finer)			Exp10-GP (Consider Finer)			Exp11-GP (Classify Finer)		
	P	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
not hate	83.96	60.65	70.42	96.36	14.94	25.86	95.65	1.66	3.25
implicit hate	39.41	43.04	41.15	27.26	47.92	34.75	27.13	65.60	38.39
explicit hate	16.77	64.68	26.63	12.26	85.32	21.44	13.99	67.89	23.20
Macro-average	**46.71**	**56.12**	**46.07**	45.29	49.39	27.35	45.59	45.05	21.61

Table 5.3: All prompts performance

Notes:

- Exp10-GP+(Consider Finer): classification over 3 major classes. General instruction prompt + "For implicit class consider" finer-level (implicit class) labels.
- Exp11-GP+(Classify Finer): classification over 9 classes and mapping back to 3 major classes. General instruction prompt to classify over all classes.
- See details in section "Qwen model results" (see Section 5.2).

Second, the one-shot setting (Exp7-GP+1shot) achieved average results compared to other prompts, but compared to the zero-shot base prompt (Exp5-GP) the model achieved an increase in performance of all classes (*see Table* 5.3).

The few-shot setting (Exp8-GP+FS) achieved the best performance compared to zero-shot and one-shot (*see Table* 5.3).

Class performance consistency

Fourth, we observed that overall performance increased with each prompt modification (from Exp5 to Exp8, *see Table* 5.3), but did not increase consistently for all classes. The best prompting method for the three major classes is a base prompt with eight training examples per label (Exp8-GP+FS *see Table* 5.3).

Performance per class did not improve consistently with each prompt: the overall macro-average F-1 score increased with each new prompt, but some classes decreased their performance. For example, adding the definition of hate speech in Exp6-GP+Def slightly improved overall performance compared to the simple instructions of Exp5-GP. The increase was mainly driven by the improvement in the F1 score for *implicit hate*, the F1 score for *not hate* decreased, and for *explicit hate* mostly remained the same (*see Table* 5.3). Adding more training examples to the base prompt (Exp8-GP+FS) led to an increase in the performance for all classes compared to all other prompts with major classes.

Structure of prompts

Fifth, if we look at the performance of prompts with finer labels (Exp 9 to Exp 11), we can conclude that adding only finer labels does not improve performance. The structure of the prompt affects performance the most. Thus, instructing to classify over the major and finer labels together (Exp11-GP+(Classify Finer), *see Prompt* 5) leads to a decrease in the performance of *not hate* compared to the similar prompt with major labels (Exp5-GP) (*see the results in Table* 5.3). Performance decreased significantly for *not hate* (recall 1.66%), but

increased for the other two classes. We observe that instructing the model to consider finer labels and indicating to which major class such finer labels belong confuses the model less. For example, the performance of all classes for Exp10-GP+(Consider Finer) is more stable compared to the Exp11-GP+(Classify Finer) (*see results in the Table* 5.3).

Training examples

Sixth, based on the results of the prompts with three major classes (Exp 5 to Exp 8), we can conclude that the best strategy to improve performance is to add more training examples. Increasing the number of training examples to eight per label, as recommended by Han and Tang (2022), increased performance by more than ten points: F1-score for One-shot prompt (32.11%) vs. F1-score for Few-shot prompt (43.42%) (*see Table* 5.3).

The combination of finer labels and few-shot settings achieved the best results in the prompt Exp9-GP+FS+(Consider Finer) (*see Table* 5.3).

Chapter 6

Error analysis

6.1 Confusion matrix review

This chapter provides a manual error analysis of the results of the best-performing prompt (Exp9-GP+FS+(Consider Finer)) (*see the results in Table* 5.3).

Among the three major labels, the model shows strong performance in identifying *not hate*: 83.96% precision and 60.65% recall (*see Table* 5.3). The *implicit hate* class reached 39.41% precision and 43.04% recall. The model struggles the most to predict the *explicit hate* class, which achieved 16.77% precision and 64.68% recall.

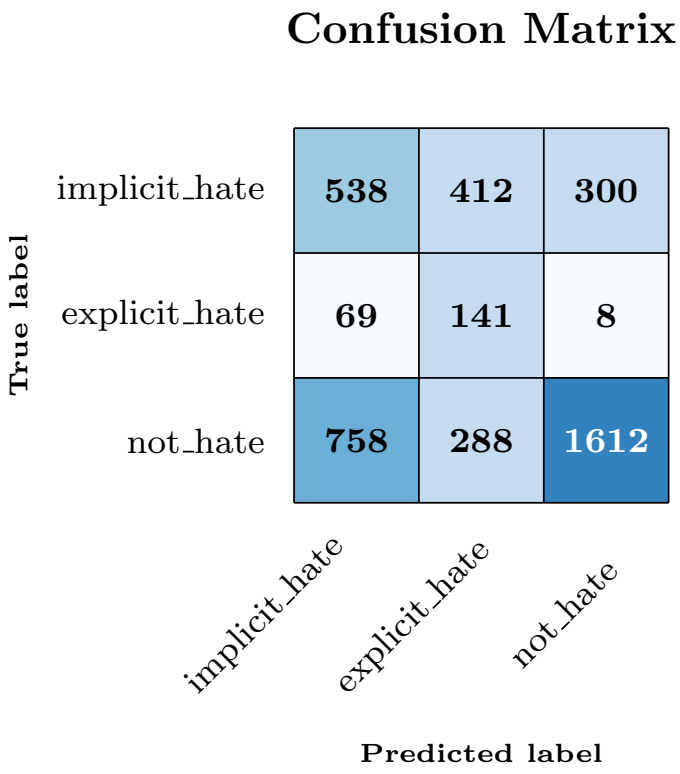

Figure 6.1: Confusion Matrix for the best performing prompt (Exp9-GP+FS+(Consider Finer)

Based on the confusion matrix overview (*see Confusion matrix* 6.1), we observe the following trends in the performance of implicit classes:

(A) The model labeled 412 *implicit hate* messages as *explicit hate*.

(B) The model labeled 758 *not hate* messages as *implicit hate*.

(C) The model labeled 288 *not hate* messages as *explicit hate*.

(D) The model labeled 69 *explicit hate* messages as *implicit hate*.

(E) The model labeled 300 *implicit hate* messages as *not hate*.

The results for (A) and (D) show that *explicit hate* and *implicit hate* are frequently confused. The model labels a message correctly as hateful, but it seems to confuse the levels of explicitness.

The results for (B) and (E) show that *implicit hate* and *not hate* are frequently confused. The model seems to struggle with *implicit hate* recognition; it may be due to the subtlety of the language.

The results for (C) are more unexpected. This could be caused by the content of the data labeled *not hate*, which could contain words or specific patterns that are often observed in the hateful data. To investigate the mentioned results, we perform a qualitative analysis and look at the patterns of each implicit class.

6.2 Qualitative error analysis of implicit classes

Implicit Class	**Total**	**Implicit%**	**Explicit%**	**Not Hate%**
White Grievance	301	56.5	29.9	13.6
Incitement	252	42.7	35.9	21.4
Stereotypical	223	48.4	35.3	16.3
Inferiority	172	38.5	24.9	36.7
Irony	156	42.9	21.2	35.9
Threatening	130	9.2	56.9	33.8
Other	16	62.5	18.8	18.8

Table 6.1: Implicit classes performance

We analyzed confusions in implicit classes to determine whether these confusions appear more or less frequently within the implicit classes defined by ElSherief et al. (2021). The predictions per implicit class can be found in the table 6.1: the column "Implicit" contains percentage per class of correct predictions, the column "Explicit" contains percentage per class of cases where the model incorrectly labeled a message as *explicit hate*, and the column "Not hate" contains percentage per class of cases where the model incorrectly labeled a message as *not hate*.

Implicit Class	**Implicit Hate**	**Explicit Hate**	**Not Hate**
White_Grievance	33	34	33
Incitement	33	34	33
Stereotypical	33	34	33
Inferiority	33	34	33
Irony	33	33	33
Threatening	12	34	33
Other	10	3	3

Table 6.2: Distribution of sampled data for error analysis

We observe that not all implicit classes achieved an equally high performance that varies from the lowest 9.2% (for *Threatening*) to the highest 56.5% (for *White Grievance*) (*see Table* 6.1).

Explicit hate	**Implicit hate**	**Not hate**
Language for all *implicit hate* classes		
	White Grievance	
Alarmist, expressing extreme violence, focusing on external threats, and conspiracy theories (23 out of 34): *"the white race is being slowly exterminated & our future is uncertain. will you take a stand today & join our efforts?"*	Ironic, mocking, expressing frustration (20 out of 33): *"i can appreciate that. i'm against non white immigration because far too many hate whites."*	Constrained, subtle (24 out of 33): *"im white and im a happy gal.. does that mean im racist and a supremist?"*
	Incitement	
Aggressive, calling for violence, sloganeering (27 out of 34): *"we must secure the existence of our people and a future for white children!"*	Coded, expressed in a form of political statement, ironic, sarcastic (25 out of 33): *"barbara spectre : muslim invasion into europe a good thing"*	Constrained, expressed in a form of political activism (25 out of 33): *"#stopvoterfraud #obam's #amnesty for #illegals #secureourborders now!!"*
	Stereotypical	
Threatening, inciting violence (29 out of 34): *"" white " " jews feel nothing 4 the lives of whites ! they only care about jews!"*	Presenting arguments as "factual", coded (26 out of 33): *"official: 75% of berlin criminals are arabs #swrm #hmm via"*	Subtle, expressing mainstream prejudice, microaggressions (24 out of 33): *"you are a blond you don't know your ass from a hole in the ground! no matter what color your hair is your blond."*
	Inferiority	
Direct aggressive calls for violence, using slurs and dehumanizing terms (29 out of 34): *"wtf is wrong with this country? this useless piece of trash comes to this country illegally then ruins the lives of 130 people disgusting"*	Using academic-sounding pseudo-scientific terms, reasoning dehumanization (19 out of 33): *"i believe it is a combination of heredity, culture, and tradition. things which blacks clearly lack. see detroit"*	Casual discrimination, conversational with subtle prejudice (27 out of 33): *"She's a girl, of course she has mood swings."*
	Irony	
Hateful, violent jokes (32 out of 33): *"I'm not racist racism is a crime, and crime is for black people."*	Borderline coded stereotyping humor (31 out of 33): *"how many feminists does it take to change a light bulb? feminists can't change anything."*	Subtle, indirect, expressing prejudice (31 out of 33): *"what pronoun should we use to refer to people who are transitioning into martians?"*
	Threatening	
Extremely violent, dehumanizing with direct calls for execution (33 out of 34): *"round them up send them home. asap."*	Aggressive, calling for action (11 out of 12): *"send them to prison and let the sisters take care of them."*	Reasoning, calling for policy-reinforcement (20 out of 33): *"cut all fedfunds to sanctuary cities. arrest mayors violating fedlaw. check their bank accounts for illegal donations."*

Table 6.3: Implicit classes linguistic patterns with message count and examples (one message can contain different patterns)

Some implicit classes were more often confused with *explicit hate* (e.g., *Threatening*, *Incitement*, *Stereotypical*), while others were more often confused with *not hate* (e.g., *Inferiority*, *Irony*). This indicates that the implicit classes possess different features. Through manual error analysis, we aimed to identify the sources of frequent disagreements and agreements for each of these implicit classes. For qualitative analysis, we investigated the predicted labels for each implicit class. We randomly selected 100 samples per class with an equal distribution of predicted data for each of the three major labels. The count of messages per class was equally distributed, except the implicit class *Threatening* that has only 12 predicted messages with the label *implicit hate*, and the implicit class *Other* that has a total count of 16 messages for all classes in dataset (*see Table* 6.2).

Through qualitative analysis of the sampled data, we identified recurring linguistic patterns and aimed to identify generalizable characteristics common to all implicit classes. Table 6.3 contains data examples with the number of messages in which a linguistic pattern was observed per implicit class. All data examples contain unrevised messages with all grammar and spelling kept as provided in the data set of ElSherief et al. (2021). In addition, we performed a lexical analysis for each implicit class to identify whether the most frequent tokens may have impacted the predictions.

6.2.1 Language pattern of *White Grievance*

The majority class *White Grievance* achieved the best result (56.5%) among classes with sufficient data. The model predicted this class as *explicit hate* in 29.9% of all cases. *White Grievance* has the lowest proportion of messages labeled *not hate* among all classes (13.6%).

In the cases where the model marked the message as *not hate* the model did not identify a subtle hate. For example, consider the test examples marked as *not hate*: "im white and im a happy gal .. does that mean im racist and a supremist?" or "#whitepeople should donate to pro-white groups". The tone and structure of the first message framed as a question express uncertainty, rather than an outright hateful statement; it seems that the model interprets it as a personal statement rather than hate speech because of the ambiguity of the message and absence of negative language. Without context, this kind of message is difficult to identify as hateful. The second example is more obvious in its implicitly hateful nature: the hashtag "#whitepeople" looks like one of the trending hashtags among white supremacists, the word "pro-white" seems to be more ambiguous, though together with the "#whitepeople" it expresses hateful speech which did not convince the model to mark the message as *implicit hate*.

Based on lexical analysis, the vocabulary contains words such as "anti-white", "whitegenocide", "racist", "genocide", "privilege", "#white genocide. We can hypothesize that the presence of certain words, commonly associated with hate groups, in particular white supremacists, helps the model classify messages as *implicit hate* and less as *explicit hate*, due to the absence of overtly explicit, derogatory words and slurs (*see Table* 3.4).

6.2.2 Language pattern of *Incitement*

The model predicted this class correctly in 42.7% of the cases. It was misclassified as *explicit hate* in 35.9% cases. In 21.4% of the cases, the model identified this class as *not hate*.

The high count of predicted *explicit hate* implies that, while the model often detected the underlying hateful nature of a post, it tended to misclassify these posts by overestimating their explicitness. For example, the following messages were classified as *explicit hate*: "happy birthday uncle adolf! how i wish you were still with us." or "fuhrer was a genius; see his analysis: adolf #hitler explains." It seems that, as with implicit class *White Grievance*, the model was triggered by certain words usually found in hateful discourse ("adolf", "#hitler", "kkk", "alt", "adolf", "whitepride") and flagged them as explicitly hateful (*see Table* 3.4).

For the *not hate*, nothing in the following messages flagged them as hateful for the model: "hungarian politicians protesting against invasion! hungary is the only normal country in europe + visegrad group" or "germany spent 20 billion on invasion in 2016 #swrm #germany #immigrants". There are no visibly hateful words in the first example. The hashtag "#immigrants" and the word "invasion" used together determined the hateful nature of the second example, which the model failed to catch.

The lexical analysis showed that the most frequent words for this pattern and the class in general are "aryan", "whitepride", "hitler", "power" (*see Table* 3.4), which can explain the high count of predicted *explicit hate* for this class (*see Table* 6.1).

6.2.3 Language pattern of *Stereotypical*

This class is one of the best performing classes with correctly labeled messages reaching 48.4%, the model labels as *explicit hate* 35.3% of messages and as *not hate* 16.3%.

The high count of messages predicted as *explicit hate* may be due to the fact that the model identified words associated with hateful content. The most frequent tokens for this class are "kill", "jewish", "juhad", "koran", "muslims", "infidel". For example, in messages labeled as *explicit hate*: "this is part of their auto-victimisation to justify the jihad! since ever!!!" or "another day another black male kills a white cop #whitelivesmatter", the model seems to recognize the words "kill", "jihad", "black", "white", hashtag "#whitelivesmatter" and possibly the exclamation mark used three times as indicators of an *explicit hate*.

The model identified 16.3% of the cases as *not hate*. It seems that in most of the cases there were no trigger keywords in the test samples. For example, the following messages are implicitly hateful because they utilize sarcasm, irony, mockery: "Wait, can I catch your Asperger's?" or "But you're not REALLY disabled. You look so pretty!" However, the model failed to detect these linguistic phenomena.

6.2.4 Language pattern of *Inferiority*

The model labeled this class as *explicit hate* 24.9% of all cases. The model incorrectly labeled this class as *not hate* (36.7%) almost as many times as *implicit hate* (38.5%).

Based on the comparison of the messages predicted as *explicit hate* with the messages predicted as *not hate* we observe that messages predicted as *explicit hate* tend to be more instigating, calling for action (e.g., "islam must be banned from all civilized societies"). Meanwhile, the messages predicted as *not hate* are mostly statements with implicitly hateful content utilizing various linguistic phenomena (e.g., "She's a girl, of course she has mood swings.").

The high rate of misclassification of *implicit hate* and *not hate* suggests that subtle and indirect expressions of inferiority are frequently perceived as non-hateful or neutral by the model, causing it to overlook them. Among the top tokens in this class are words like "animal," "rat," "savage," and "parasite," whose hateful nature is highly context-dependent (*see Table* 3.4). The test samples like "You might want to think about losing weight if you want to get married and have children. Men are very visual creatures." or "one monkey don't stop no show" do not trigger the model to mark them as hateful. In both cases, the model struggles with subtle, coded language; while the first example is reinforcing gender stereotyping, body-shaming, and misogyny, the second message is more ambiguous, there is no obvious target, and it is hard to label it as hateful without the context.

We observe that the use of slurs and direct violent targeting intensified in the messages predicted as *explicit hate* ("trash", "shit", "subhuman"), there are less derogatory words in the messages predicted as *implicit hate* ("savages", "idiots", "cockroaches"), and only a few ("vermin", "imbicile", "slime", "animals") in the messages predicted as *not hate*.

6.2.5 Language pattern of *Irony*

The class was predicted as *explicit hate* for 21.2% of messages. Among the implicit classes with a comparable amount of data, this class represents the lowest proportion of messages labeled *explicit hate*. This may be attributed to the absence of overtly hateful words among the top-ranked tokens in this class, in contrast to other implicit classes (e.g., *Inferiority*) where such language is more prominent (*see Table* 3.4). The model labeled as *implicit hate* 42.9% and as *"not hate"* 35.9% of the messages.

We analyzed messages marked as *explicit hate*: 22 out of 33 messages contained the words "jews", "black", "supremacist", "swastika", "hitler", "kill". We conducted the same analysis of the messages marked as "not hate" and found that 3 out of 33 messages contained these words. We observe that the model seems to be triggered by the words that are frequently linked to hateful content. The fact that only a small proportion of messages contain such words confirms the high count of *not hate*. For example, there is no obviously abusive language in messages labeled as *not hate*, the intent to hurt is subtle and ironic: "how dare you assume gender. it's equality, baby." or "if you skip class to protest trump you might be a college dropout soon anyway!"

6.2.6 Language pattern of *Threatening*

The implicit class *Threatening* was labeled as *explicit hate* more than any other implicit class (56.9%); and as *implicit hate* less than any other implicit class (9.2%). The model labeled 33.8% of the messages as *not hate*.

The model classified more than half of the cases as *explicit hate*. Although the most frequent tokens (e.g., "deport", "send", "ice" (*see Table* 3.4) for this class are not as explicitly violent as the most frequent tokens of the implicit class *Stereotypical*, for example, the messages in the sampled data that were incorrectly labeled as *explicit hate*, contain verbs closely associated with the language inciting violence ("shoot", "lock", drive", "screw", etc.). Most of the sampled messages labeled as *explicit hate* contain some kind of call for action (e.g., "ban hijab and burka. they need to assimilate or leave"); include exclamation marks along with inciteful language, which likely made it easier for the model to classify them as *explicit hate*. At the same time, we observe that *Threatening* is one of the three classes that has the highest count of messages incorrectly labeled as *not hate* (*see Table* 6.1). The other two classes are *Inferiority* and *Irony*. A closer examination of the messages predicted as *not hate* reveals the absence of explicitly hateful language, the verbs commonly observed in the messages predicted as *explicit hate* appear less frequently in this group. Nevertheless, some messages remain inciteful in nature (e.g., "let's end sanctuary cities for good!!! americans come first. let illegals come in the rig").

6.2.7 Language pattern of *Other*

The class *Other* has the smallest count of test data, the model correctly labeled 62.5% of it. 18.8% of the messages were labeled as *explicit hate* and 18.8% as *not hate*. The model correctly labeled messages as *implicit hate* when they conveyed a subtle hateful connotation without using extremely aggressive language (e.g., "Why can't all Black Women wear their hair like yours?"). The messages marked as *explicit hate* contained slurs and clearly hateful words (e.g., "jared taylor is an #antiwhite #cuckservative and his supporters are traitorous trash"). At the same time, the message with the same slur was marked as *not hate* (e.g., "the federalist : please stop glorifying manufacturing jobs via #globalization #freetrade #cuckservative"). It seems that the model did not recognize all derogatory terms and was triggered by other likely more frequent hateful speech words (e.g., "#antiwhite", "trash").

6.3 Summary

We conducted a qualitative analysis of the predicted messages. Based on our observations, we can conclude the following.

We observed that some implicit classes were more often confused with *explicit hate* (e.g., *Threatening*, *Incitement*, *Stereotypical*), while others were more often confused with *not hate* (e.g., *Inferiority*, *Irony*). Upon closer analysis, we conclude that the model identifies patterns in the language used within messages and leverages these as workarounds to predict a certain label. Therefore, the language and vocabulary of the most frequent words impact the prediction the most.

1. The model tends to label a message as *explicit hate* if it observes aggressive, extremely violent, clearly hateful language, slurs (*see Table* 6.3). For example, the implicit classes with the highest number of messages predicted as *explicit hate*, *Threatening*, *Incitement*, *Stereotypical*, contain easily identifiable words commonly associated with hateful content. This includes explicit terms such as "kill," "jihad," "#hitler," "kkk," "alt," and "adolf" within the implicit classes *Incitement* and *Stereotypical*, as well as verbs linked to violent incitement such as "shoot", "lock", drive", "screw" within the implicit class *Threatening*. The presence of such language commonly associated with hateful comments appears to trigger the model to classify these messages as *explicit hate*.

2. The model frequently misclassifies messages as *not hate* due to difficulty in detecting subtle and constrained language (e.g., in implicit class *Incitement*), particularly when dehumanizing ideas are expressed in a casual, constrained tone. The model is more likely to label as *not hate* those messages that contain subtle hateful language, including microaggressions, subtle prejudice, and casual discrimination that is often masked by references to policy reinforcement or framed within the context of political activism (e.g., in implicit class *Stereotypical*) (*see Table* 6.3).

The classes with the least number of messages labeled as *not hate* are *White Grievance*, *Stereotypical*, *Incitement*, *Other*. And if we observe that such implicit classes as *Stereotypical*, *Incitement* operate with strong, explicitly hateful language, implicit class *White Grievance* operates with words commonly associated with white supremacy and common U.S. hate groups (*see Tables* 6.1, 3.4). The model more often labeled as *not hate* classes whose language is less derogatory, explicitly hateful, and extremely violent. For example, implicit classes with the least distinctively hateful frequent words in their vocabulary (*Inferiority*, *Irony*, *Threatening*) have the highest number of messages labeled as *not hate* (*see Tables* 6.1, 3.4). The model predicts *not hate* when a message contains a combination of different words that are not hateful by themselves, but create a hateful connotation together (e.g., hashtag "#immigrants" used together with the word "invasion" in the same message).

Except for the implicit class *White Grievance*, the model struggles to identify ironic, mocking, and sarcastic language, and the model predicts such messages as *not hate* (e.g., in the class *Stereotypical*) (*see Section* 6.2).

3. The model correctly labels a message as *implicit hate* if it observes borderline coded humor (e.g., for implicit class *Irony*), academic-sounding language combined with reasoning dehumanization (e.g., for implicit class *Inferiority*) (*see Table* 6.3).

It should be noted that the implicit class *Threatening* shows the lowest performance for *implicit hate* prediction at 9.2%, whereas the *White Grievance* class achieves the highest performance at 56.5% (*see Table* 6.1). A commonality between these two classes is the absence of highly explicit or overtly derogatory top tokens (*see Table* 3.4). However, the model appears better equipped to identify as *implicit hate* the *White Grievance* class. This may be due to the contrast in linguistic expression in this class: messages predicted as *implicit hate* tend to employ irony, mockery, and frustration in combination with the tokens associated with white supremacy, while messages predicted as *explicit hate* are more alarmist

and overtly violent. In contrast, the *Threatening* class possesses greater challenges for the model. The messages of this implicit class are more homogeneous, often include calls for action, and operate with many different verbs. As a result, the model frequently misclassifies messages as either *explicit hate* or *not hate* based on the level of violence and dehumanization of the language, while failing to predict *implicit hate* in most messages.

4. Overall, the model detects most of the implicit classes' messages as hateful (*see Table* 5.3).

The factors that lead to classification of a message as *explicit hate* or *implicit hate* vary (*see Section* 6.2). At the same time, in the actual application of hate speech detection, the distinction between *explicit hate* and *implicit hate* would not appear to cause an issue as long as hateful content is detected.

The fact that the model struggled to distinguish the level of explicitness of the language may be because the model's "understanding" of *implicit hate* differs from the one adopted for annotation in the dataset by ElSherief et al. (2021), which we used for the study.

Chapter 7

Discussion and Conclusion

7.1 Overview

The goal of our thesis was to evaluate a generative model performance by instructing it with various prompts. In particular, our research questions were:

Research questions

1) Can generative model combined with prompting techniques outperform masked language models like BERT in detecting implicit hate speech?

2) What prompting techniques are the most effective in improving the detection of implicit hate speech?

Sub-questions

1a) What is the performance of BERT on this dataset?

1b) What is the performance of the generative model on this dataset?

2a) How do different prompting techniques affect the results for implicit hate speech detection?

2b) How does incorporating external knowledge about hate speech into the prompts impact performance on implicit hate speech detection?

To address these questions, we selected the implicit hate dataset, which was specifically designed for the detection of implicit hate speech (ElSherief et al., 2021). Using this dataset, we conducted a series of experiments aimed at answering these research questions and sub-questions.

Based on the results of our experiments with generative and masked language models, under the prompting strategies employed in this study, the generative language model Qwen failed to outperform the masked language model BERT in detecting implicit hate speech. On our data set, the BERT-base-uncased model achieved a macro-average F-1 score of 66.67%, precision 66.20%, recall 67.14% for *implicit hate* (*see Section* 5.1). On the same data set, the best performing prompt for the Qwen achieved a macro-average F-1 score 41.15%, precision 39.41%, recall 43.04% for *implicit hate* (*see Section* 5.2).

The best performing prompt is Exp9-GP+FS+(Consider Finer). In this prompt, we provided eight training examples per each major class (*explicit hate*, *implicit hate*, *not hate*) in the prompt and instructed to classify across the three major labels and for *implicit hate* to consider six implicit class labels excluding *Other* (*see Section* 4.6).

To answer Question 1a), we conducted experiments with three masked language models: BERT-base-uncased model, HateBERT, and RoBERTa. The best performing model for our dataset was the BERT-base-uncased. To know whether the BERT model could benefit from classification in fine-grained implicit classes, we used this model to conduct an experiment with finer labels when the model was fine-tuned with nine classes, and classification was done across three major classes. The overall performance of the finer labels BERT-base-uncased was slightly worse than the performance of BERT-base-uncased.

On our dataset, the BERT-base-uncased model achieved overall macro-average F-1 score 63.91%, precision 66.99%, recall 62.01%. For *explicit hate* macro-average F-1 score is 40.74%, precision 51.16%, recall 33.85%. For *implicit hate* macro-average F-1 score is 66.67%, precision 66.20%, recall 67.14%. For *not hate* macro-average F-1 score is 84.33%, precision 83.62%, recall 85.05% (*see Table* 5.1).

To answer Question 1b), we applied various prompting methods on the generative model Qwen. On our dataset, the best performing prompt for the Qwen achieved overall macro-average F-1 score 46.07%, precision 46.71%, recall 56.12%; the results are lower than the results of the BERT-base-uncased model. For *explicit hate*, the model achieved a macro-average F-1 score 26.63%, precision 16.77%, recall 64.68%. For *implicit hate*, the model achieved a macro-average F-1 score 41.15%, precision 39.41%, recall 43.04%. For *not hate*, the model achieved a macro-average F-1 score 70.42%, precision 83.96%, recall 60.65% (*see Table* 5.3).

To answer Question 2a), we conducted experiments with various prompting methods (e.g. incorporating the hate speech definitions, training examples, finer labels to the prompt) (*see Section* 4.6). We implemented an eight-shot prompt described by Han and Tang (2022) and followed their general guidelines on prompting with additional fine-grained labels. The structure of the prompt helped the model to perform better compared to other prompts that utilized finer labels. Incorporating finer labels in the prompt also helped to improve the results. In general, we observed that the definitions of hate speech and the training examples in the prompt increased its macro-average F-1 score (*see Table* 5.3). We identified that Qwen performed the best when asked to classify into three major classes with consideration of the finer labels in the few-shot setting (*see Table* 5.3).

To answer Question 2b), we incorporated the hate speech definitions into the prompts (*see Section and the prompt example* 4.6, 1). Adding hate speech definitions improved the performance of implicit hate speech detection compared to the base prompt (general instructions to classify a message).

Our manual qualitative error analysis revealed that the language of the implicit classes impacted the performance the most. Some implicit classes were incorrectly predicted as *explicit hate* and others as *not hate* more frequently than other classes (*see Section* 6.3). Implicit class messages misclassified as *not hate* often contained subtle, constrained language that often expressed microaggressions, subtle prejudice, casual discrimination, and such linguistic phenomena as irony and sarcasm.

Regarding *explicit hate* prediction, we raise several concerns about the data set developed by ElSherief et al. (2021). First, implicit class messages predicted as *explicit hate* often contain intense, aggressive language, including dehumanizing remarks or incitement to violence. We observe that the implicit classes vocabulary often contains words that are more associated with *explicit hate* (e.g., "kill", "kkk", "whitepride") and various slurs. We conclude that the occurrence of explicitly hateful terms typically associated with *explicit hate* contributed to the model's systematic misclassification of these messages as *explicit hate* (*see Table* 3.4).

Second, the model appears to classify inciteful calls to action as *explicit hate*. For example, implicit class *Threatening* achieved the lowest performance in detecting *implicit hate*, meanwhile classifying more than half of the messages as *explicit hate*. This raises the question of whether this class should be included in the *implicit hate* data set in the first place and to what extent messages of this type can be appropriately labeled as *implicit hate*.

Third, we also observe that the messages are often very similar in their hateful nature. For instance, most messages within the implicit class *Threatening* contain a call to action and, therefore, could also be labeled as *Incitement*.

This leads us to the conclusion that the data set developed by ElSherief et al. (2021), and in particular the messages labeled as *implicit hate*, do not fully align with the model's learned

representation of *implicit hate*. Furthermore, we conclude that the inclusion of explicitly hateful messages in the dataset and labeling them as *implicit hate* is inconsistent with the definition of *implicit hate* provided by the creators of the dataset, who define implicit hate speech as "a coded or indirect language such as sarcasm, metaphor, and circumlocution used to disparage a protected group or individual, or to convey prejudicial and harmful views about them." (ElSherief et al., 2021).

The aforementioned issues with the dataset, which was specifically developed for implicit hate detection, underscore a broader problem: the difficulty of developing a comprehensive dataset for *implicit hate*. The results suggest that, while the generative model is capable of detecting hate speech, it is often confounded by the explicit nature of the language present in the dataset. A more refined data set, developed in closer alignment with the nature of implicit hate, could potentially improve the performance of the generative model.

7.2 Limitations and future work

The experiment with the Qwen model parameters was non-systematic and did not cover the whole development data set.

This study was primarily based on a single data set and involved a limited number of experiments; therefore, it remains uncertain whether the results generalize to other data sets or experimental settings.

The annotation of implicit hate classes implemented in the data set raised some concerns (*see Section* 7.1). In real-world applications of automated hate speech detection, the importance of distinguishing between *explicit* and *implicit hate* remains an open question. In practice, identifying *implicit hate* may not be a primary objective as long as hateful messages are generally detected. And overall, closer attention should be paid to the distinction between *implicit hate* and *not hate*. The generative model Qwen demonstrated the ability to detect hateful messages (*see Table* 5.3), raising the question whether researchers should prioritize general hate detection over fine-grained distinctions between *explicit hate* and *implicit hate*. In this context, if one of the research focuses is the distinction between various hate types, then the removal of *explicit hate*, as undertaken by ElSherief et al. (2021) (*see Section* 3.1), does not appear to contribute to the research of hate speech.

The data set used for the study focused specifically on *implicit hate*, but also contained *explicit hate* and *not hate* labels. Some of the *implicit hate* classes utilized the words often associated with *explicit hate* (*see results* 6.3). For future work, it can be useful to work with other *implicit hate* datasets that cover various classes and linguistic phenomena common to *implicit hate*. The removal of explicitly hateful messages could help redirect focus toward *implicit hate* that more closely reflects its true nature: subtle, coded, and indirect expressions of hate that may also use irony, sarcasm, metaphor, etc.

The data set included the implicit class *Other*, which we considered confusing for the model and removed in the prompts that asked to "consider" implicit class labels when classifying across three major labels. Possibly incorporating more *implicit hate* classes into the prompt could help improve performance. Due to the nature of *implicit hate*, there is a lack of various comprehensive data sets that could be classified and compared across. Context plays a crucial role in *implicit hate* detection we used the data set with standalone messages, which the model often struggled to label as *implicit hate*. Incorporation of previous threads of posts could help the model detect patterns in *implicit hate* expression.

Appendix

Exp 5 - GP (General prompt)	Classify the following comment by responding with ONLY ONE of these exact labels and nothing else: not_hate explicit_hate implicit_hate Respond ONLY with one of the above labels.
Exp 6 - GP + Def (General prompt with hate speech definition)	Explicit hate speech is any communication that disparages a person or a group on the basis of characteristic such as race, color, ethnicity, gender, sexual orientation, nationality, religion, or other characteristics. Implicit hate speech is a coded or indirect language such as sarcasm, metaphor, circumlocution, etc. used to disparage a protected group or individual, or to convey prejudicial and harmful views about them. Classify the following comment by responding with ONLY ONE of these exact labels and nothing else: not_hate explicit_hate implicit_hate Respond ONLY with one of the above labels.
Exp 7 - GP + 1shot (One-shot prompt (one training example per label)	Classify the following comment by responding with ONLY ONE of these exact labels and nothing else: not_hate explicit_hate implicit_hate Comment: "sure! whatever whitey!" implicit_hate Comment: "white south africans trapped like frogs in boiling water as anti-white racial violence escalates #swrm" explicit_hate Comment: "white people founded & build south africa and have been there since 1600s." not_hate # Respond ONLY with one of the below labels: IMPORTANT: Your entire response must be EXACTLY ONE of these three labels with no explanation, no reasoning, and no additional text: not_hate explicit_hate implicit_hate

Table 1: Major labels prompts (Exp 5 to Exp 7)

Exp 8 - GP + FS (Few-shot prompt)	Classify the following comment by responding with ONLY ONE of these exact labels and nothing else: not_hate explicit_hate implicit_hate Comment: "sure! whatever whitey!" implicit_hate (*8 training examples in total*) Comment: "hurrrr hurrrrr white trash power" explicit_hate (*8 training examples in total*) Comment: "white people founded & build south africa and have been there since 1600s." not_hate (*8 training examples in total*) Respond ONLY with ONE of these EXACT labels: IMPORTANT: Your entire response must be EXACTLY ONE of these labels with no explanation, no reasoning and no additional text. not_hate explicit_hate implicit_hate

Table 2: Exp 8 - GP + FS (Few-shot prompt)

Exp 9 - GP + FS + (Consider Finer)	Comment: “sure! whatever whitey!” implicit_hate (*8 training examples in total*) Comment: “hurrrr hurrrrr white trash power” explicit_hate (*8 training examples in total*) Comment: “white people founded & build south africa and have been there since 1600s.” not_hate (*8 training examples in total*) Classify the comment into ’explicit_hate’, ’implicit_hate’ or ’not_hate’. For ’implicit_hate’ consider if it is ’white_grievance’, ’incitement’, ’stereotypical’, ’inferiority’, ’irony’, ’threatening’. Comment: post Respond ONLY with ONE of these EXACT labels: not_hate explicit_hate implicit_hate IMPORTANT: Your entire response must be EXACTLY ONE of these labels with no explanation, no reasoning and no additional text.””””

Table 3: Exp 9 - GP + FS + (Consider Finer)

Exp 10 - GP + (Consider Finer)	Classify the following comments into ’explicit_hate’, ’implicit_hate’ or ’not_hate’. For ’implicit_hate’ consider if it is ’white_grievance’, ’incitement’, ’stereotypical’, ’inferiority’, ’irony’, ’threatening’, ’other’. Respond ONLY with ONE of these EXACT labels: white_grievance incitement stereotypical inferiority irony threatening other explicit_hate not_hate IMPORTANT: Your entire response must be EXACTLY ONE of these labels with no explanation, no reasoning and no additional text.””””

Table 4: Exp 10 - GP + (Consider Finer)

Exp 11 - GP + (Classify Finer)	Classify the following comments into "white_grievance', 'incitement', 'stereotypical', 'inferiority', 'irony', 'threatening', 'other' or 'explicit_hate', 'not_hate'. Respond ONLY with ONE of these EXACT labels: white_grievance incitement stereotypical inferiority irony threatening other explicit_hate not_hate IMPORTANT: Your entire response must be EXACTLY ONE of these labels with no explanation, no reasoning and no additional text.

Table 5: Exp 11 - GP + (Classify Finer)

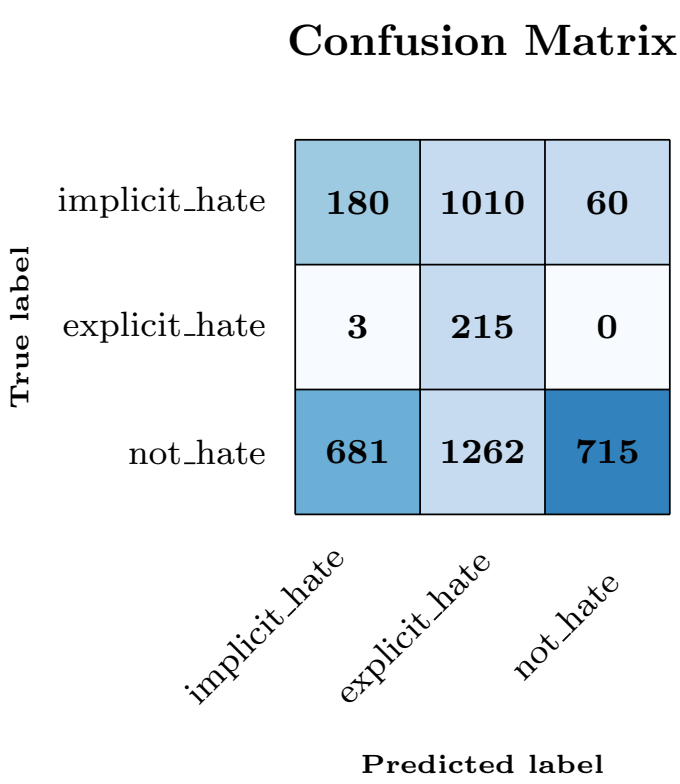

Figure 1: Confusion Matrix for the prompt Exp 5 - GP (General prompt)

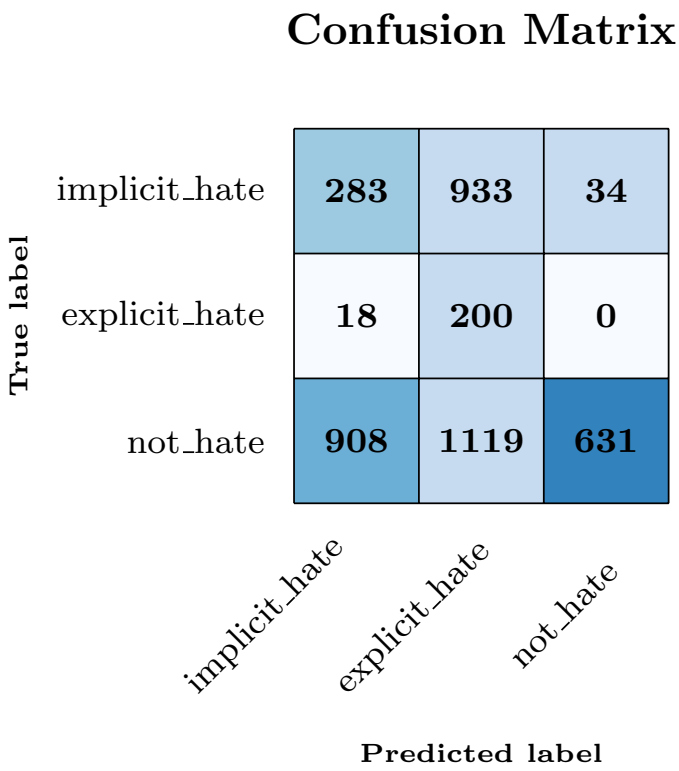

Figure 2: Confusion Matrix for the prompt Exp 6 - GP + Def

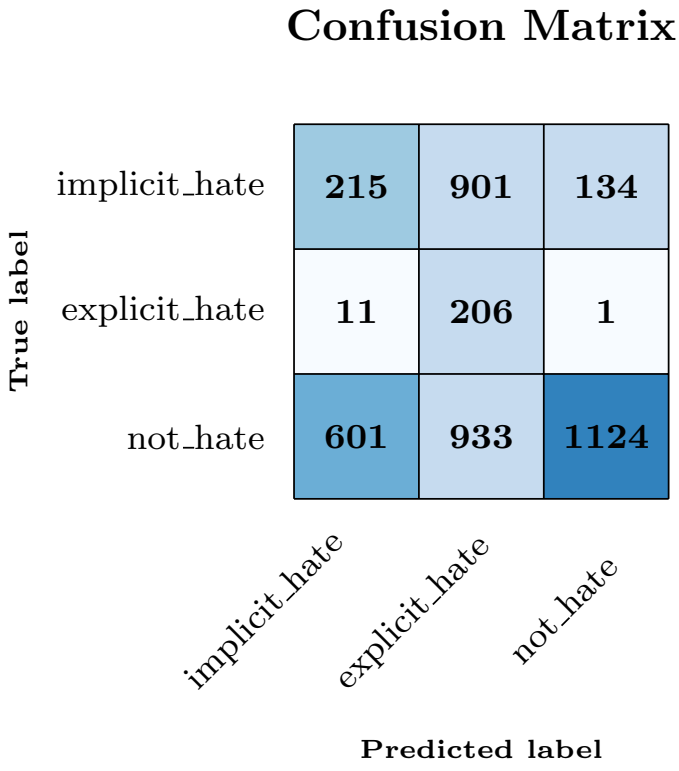

Figure 3: Confusion Matrix for the prompt Exp 7 - GP + 1shot

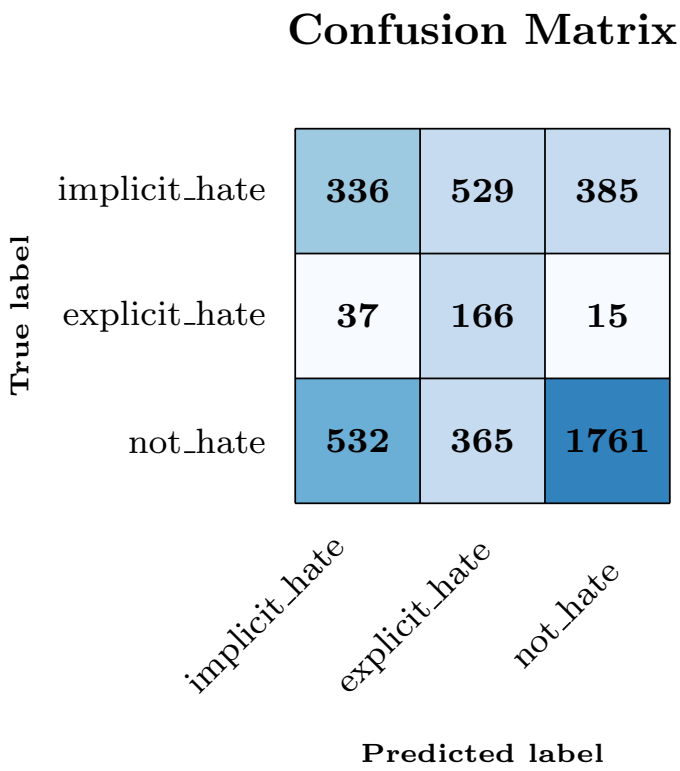

Figure 4: Confusion Matrix for the prompt Exp 8 - GP + FS

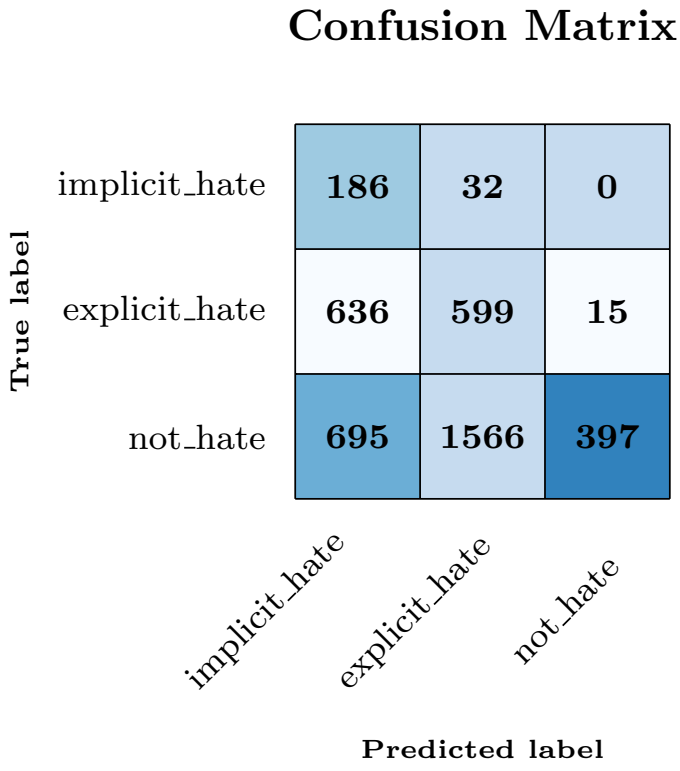

Figure 5: Confusion Matrix for the prompt Exp10 - GP + (Consider Finer)

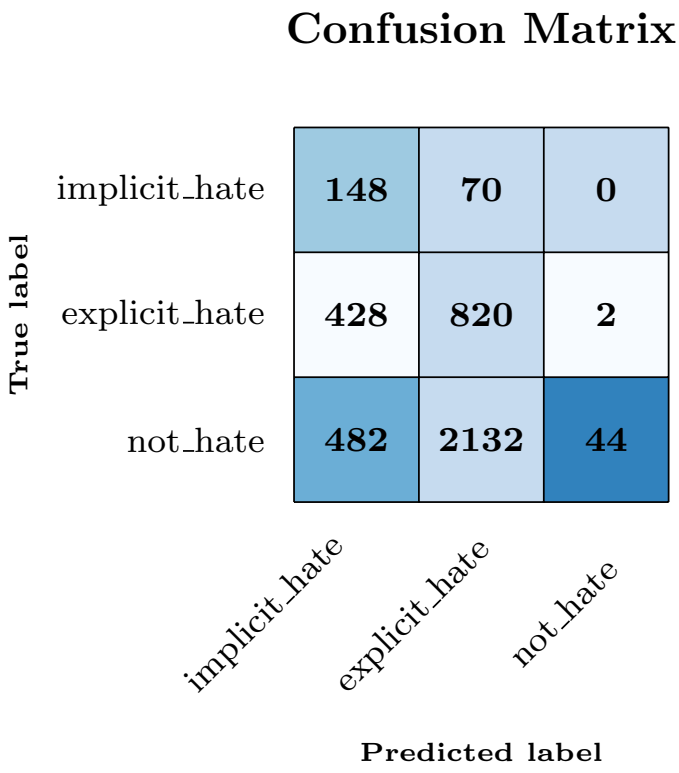

Figure 6: Confusion Matrix for the prompt Exp11 - GP + (Classify Finer)

References

P. Alonso, R. Saini, and G. Kovács. Thenorth at semeval-2020 task 12: Hate speech detection using roberta. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation (SemEval-2020)*, pages 2197–2202, Barcelona, Spain (Online), 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.semeval-1.292. URL https://aclanthology.org/2020.semeval-1.292/.

N. A. Alzahrani, H. A. Alyahya, Y. Alnumay, S. Alrashed, S. Z. Alsubaie, Y. Almushayqih, F. A. Mirza, N. M. Alotaibi, N. Al-Twairesh, A. Alowisheq, M. S. Bari, and H. Khan. When benchmarks are targets: Revealing the sensitivity of large language model leaderboards. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13787–13805, Bangkok, Thailand, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.acl-long.744. URL https://aclanthology.org/2024.acl-long.744/.

M. Berbrier. The victim ideology of white supremacists and white separatists in the united states. *Sociological Focus*, 33(2):175–191, 2000.

T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html.

T. Caselli, V. Basile, J. Mitrović, I. Kartoziya, and M. Granitzer. 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 (LREC 2020)*, pages 6193–6202, Marseille, France, 2020. European Language Resources Association. URL https://aclanthology.org/2020.lrec-1.760.

T. Caselli, V. Basile, J. Mitrović, and M. Granitzer. 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. Association for Computational Linguistics, 2021. doi: 10.18653/v1/2021.woah-1.3. URL https://aclanthology.org/2021.woah-1.3.

X. Chen, Y. Cui, Q. Ye, S. Chen, Z. Yang, Y. Zhu, H. Liu, Y. Luo, J. Lin, Z. Liu, and J. Yang. Mixture-of-instructions: Aligning large language models via mixture prompting. *arXiv preprint arXiv:2404.18410*, 2024.

A. Cloud. Alibaba cloud model studio: Qwen api reference. https://www.alibabacloud.com/help/en/model-studio/use-qwen-by-calling-api, 2025. Accessed July 23, 2025; details OpenAI-compatible parameters including 'max_tokens' behavior.

M. de la Vega. Understanding openai's "temperature" and "top_p" parameters in language models, Nov. 2023. URL https://medium.com/@1511425435311. Accessed: 2025-05-06.

J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL https://aclanthology.org/N19-1423/.

T. Dunning. Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, 19(1):61–74, 1993.

M. ElSherief, C. Ziems, D. Muchlinski, V. Anupindi, J. Seybolt, M. De Choudhury, and D. Yang. Latent hatred: A benchmark for understanding implicit hate speech. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 345–363. Association for Computational Linguistics, Nov. 2021. URL https://aclanthology.org/2021.emnlp-main.29.

L. Gao. Detecting online hate speech using context aware models. In *Proceedings of the International Conference Recent Advances in Natural Language Processing (RANLP 2017)*, pages 260–266, Varna, Bulgaria, 2017. INCOMA Ltd. doi: 10.26615/978-954-452-049-6_036. URL https://aclanthology.org/R17-1036/.

T. Gao, A. Fisch, and D. Chen. Making pre-trained language models better few-shot learners. 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 3816–3830, Online, Aug. 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.295. URL https://aclanthology.org/2021.acl-long.295/.

S. Ghosh, M. Suri, P. Chiniya, U. Tyagi, S. Kumar, and D. Manocha. Cosyn: Detecting implicit hate speech in online conversations using a context synergized hyperbolic network. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6159–6173, Singapore, 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.377. URL https://aclanthology.org/2023.emnlp-main.377/.

K. Guo, A. Hu, J. Chen, Q. Zhang, X. Shen, and H. Hu. An investigation of large language models for real-world hate speech detection. In *2023 22nd IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 1303–1309. IEEE, Dec. 2023. doi: 10.1109/ICMLA58977.2023.00237. URL https://doi.org/10.1109/ICMLA58977.2023.00237.

L. Han and H. Tang. Designing of prompts for hate speech recognition with in-context learning. In *2022 International Conference on Computational Science and Computational Intelligence (CSCI)*, pages 386–391. IEEE, 2022. doi: 10.1109/CSCI58124.2022.00063. URL https://doi.org/10.1109/CSCI58124.2022.00063.

Hatebase. Hatebase: Structured, multilingual, usage-based hate speech repository. https://hatebase.org, 2025. Launched 2013; accessed June 14, 2025.

Hugging Face. Transformers - text generation - generationconfig, 2024. URL https://huggingface.co/docs/transformers/main/en/main_classes/text_generation#transformers.GenerationConfig. Accessed: 2025-04-23.

Hugging Face. RoBERTa Model Documentation, 2025. URL https://huggingface.co/docs/transformers/en/model_doc/roberta. Accessed: 2025-05-14.

D. Jones. Noswearing.com dictionary of swear words, 2020. URL `https://www.noswearing.com/dictionary`. Accessed: 2025-06-14.

D. Jurafsky and J. H. Martin. *Speech and Language Processing*. Pearson, 2024. Copyright © 2024. All rights reserved.

D. Jurgens, L. Hemphill, and E. Chandrasekharan. A just and comprehensive strategy for using nlp to address online abuse. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3658–3666, Florence, Italy, 2019. Association for Computational Linguistics.

R. Justo, T. Corcoran, S. M. Lukin, M. Walker, and M. I. Torres. Extracting relevant knowledge for the detection of sarcasm and nastiness in the social web. *Knowledge-Based Systems*, 69:124–133, 2014.

B. Kennedy, M. Atari, A. M. Davani, L. Yeh, A. Omrani, Y. Kim, K. Coombs Jr, S. Havaldar, G. Portillo-Wightman, E. Gonzalez, et al. The gab hate corpus: A collection of 27k posts annotated for hate speech. PsyArXiv, July 2018. URL `https://doi.org/10.31234/osf.io/hq3kw`.

H. R. Kirk, W. Yin, B. Vidgen, and P. Röttger. Semeval-2023 task 10: Explainable detection of online sexism. In *Proceedings of the 17th International Workshop on Semantic Evaluation (SemEval-2023)*, pages xx–xx, Toronto, Canada, 2023. Association for Computational Linguistics.

R. Kumar. Benchmarking aggression identification in social media. In R. Kumar, A. K. Ojha, M. Zampieri, and S. Malmasi, editors, *Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying (TRAC-2018)*, pages 1–11, Santa Fe, New Mexico, USA, August 2018. Association for Computational Linguistics. URL `https://aclanthology.org/W18-4401/`.

G. Lei, K. Alexis, and H. Ruihong. Recognizing explicit and implicit hate speech using a weakly supervised two-path bootstrapping approach. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 774–782, Taipei, Taiwan, 2017. Asian Federation of Natural Language Processing. URL `https://aclanthology.org/I17-1078`.

L. Li, L. Fan, S. Atreja, and L. Hemphill. "hot" chatgpt: The promise of chatgpt in detecting and discriminating hateful, offensive, and toxic comments on social media. 2023. Preprint.

P. Liu, W. Yuan, J. Fu, Z. Jiang, H. Hayashi, and G. Neubig. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys (CSUR)*, 55(9):1–35, 2023. doi: 10.1145/3533375. URL `https://doi.org/10.1145/3533375`.

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

Y. MacAvaney, Sean. Hate speech detection: Challenges and solutions. *PLOS ONE*, 14(8): e0221152, 2019. doi: 10.1371/journal.pone.0221152. URL `https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0221152`.

MetaAI. Llama 3.1 model cards and prompt formats, jul 2024. URL `https://www.llama.com/docs/model-cards-and-prompt-formats/llama3_1/`. Accessed: 2025-05-11.

M. Mosbach, M. Andriushchenko, and D. Klakow. On the stability of fine-tuning bert: Misconceptions, explanations, and strong baselines. In *Proceedings of the 9th International Conference on Learning Representations (ICLR)*, 2021. URL https://arxiv.org/abs/2006.04884.

R. S. Neilsen. 'toxification' as a more precise early warning sign for genocide than dehumanization? an emerging research agenda. *Genocide Studies and Prevention: An International Journal*, 9(1):9–19, 2015.

J. Nockleby. Hate speech. In *Encyclopedia of the American Constitution*. Macmillan, New York, 2000.

NVIDIA. Nvidia nim api reference: Qwen2-7b-instruct inference. https://docs.api.nvidia.com/nim/reference/qwen-qwen2-7b-instruct-infer, 2025. Specifies 'max_tokens' parameter: integer, 1–4096, default = 1024; for Qwen2-7B-Instruct model.

N. B. Ocampo, E. Sviridova, E. Cabrio, and S. Villata. An in-depth analysis of implicit and subtle hate speech messages. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1989–2005, Dubrovnik, Croatia, 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.eacl-main.143. URL https://aclanthology.org/2023.eacl-main.143.

F. Petroni, T. Rocktäschel, S. Riedel, P. Lewis, A. Bakhtin, Y. Wu, and A. Miller. Language models as knowledge bases? 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 2463–2473. Association for Computational Linguistics, 2019. URL https://aclanthology.org/D19-1250.

QwenTeam. Qwen3, Apr. 2025. URL https://github.com/QwenLM/Qwen3. Accessed: 2025-05-11.

B. Ross, M. Rist, G. Carbonell, B. Cabrera, N. Kurowsky, and M. Wojatzki. Measuring the reliability of hate speech annotations: The case of the european refugee crisis. In S. Dipper, editor, *NLP4CMC III: 3rd Workshop on Natural Language Processing for Computer-Mediated Communication*, volume 17 of *Bochumer Linguistische Arbeitsberichte*, pages 6–9, Bochum, North Rhine-Westphalia, Germany, 2016. Ruhr-Universität Bochum. URL https://www.linguistics.ruhr-uni-bochum.de/forschung/arbeitsberichte/17.pdf. Presented at the 3rd Workshop on Natural Language Processing for Computer-Mediated Communication / Social Media, 22 September 2016.

B. P. Sahoo Nihar, Gupta Himanshu. Detecting unintended social bias in toxic language datasets. In *Proceedings of the 26th Conference on Computational Natural Language Learning (CoNLL)*, pages 132–143, Abu Dhabi, United Arab Emirates (Hybrid), 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.conll-1.10. URL https://aclanthology.org/2022.conll-1.10/.

M. Sanguinetti, F. Poletto, C. Bosco, V. Patti, and M. Stranisci. An italian twitter corpus of hate speech against immigrants. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan, 2018. European Language Resources Association (ELRA).

K. Somerville. Violence, hate speech and inflammatory broadcasting in kenya: The problems of definition and identification. *Ecquid Novi: African Journalism Studies*, 32(1):82–101, 2011. doi: 10.1080/02560054.2011.545568.

D. W. Sue. *Microaggressions in Everyday Life: Race, Gender, and Sexual Orientation*. Wiley, Hoboken, NJ, 2010.

United Nations. What is hate speech?, 2025a. URL `https://www.un.org/en/hate-speech/understanding-hate-speech/what-is-hate-speech`. Accessed: 2025-04-16.

United Nations. International human rights law, 2025b. URL `https://www.un.org/en/hate-speech/united-nations-and-hate-speech/international-human-rights-law`. Accessed: 2025-04-16.

B. Vidgen. 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, 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.182. URL `https://aclanthology.org/2021.naacl-main.182/`.

W. Warner and J. Hirschberg. Detecting hate speech on the world wide web. In *Proceedings of the Second Workshop on Language in Social Media*, pages 19–26, Montréal, Canada, 2012. Association for Computational Linguistics.

Z. Waseem. Understanding abuse: A typology of abusive language detection subtasks. In *Proceedings of the First Workshop on Abusive Language Online*, pages 78–84. Association for Computational Linguistics, 2017. URL `https://aclanthology.org/W17-3012/`.

L. Xu, J. Xeng, and S. Chen. yasuo at hasoc2020: Fine-tune xlm-roberta for hate speech identification. In T. Mandl, S. Modha, G. K. Shahi, A. K. Jaiswal, D. Nandini, D. Patel, P. Majumder, and J. Schäfer, editors, *Working Notes of FIRE 2020 - Forum for Information Retrieval Evaluation*, volume 2826, pages 311–318. CEUR-WS.org, 2020. URL `https://ceur-ws.org/Vol-2826/`. Accessed: 2025-05-14.

A. Yang, B. Yu, C. Li, D. Liu, F. Huang, H. Huang, J. Jiang, J. Tu, J. Zhang, J. Zhou, J. Lin, K. Dang, K. Yang, L. Yu, M. Li, M. Sun, Q. Zhu, R. Men, T. He, W. Xu, W. Yin, W. Yu, X. Qiu, X. Ren, X. Yang, Y. Li, Z. Xu, and Z. Zhang. Qwen2.5-1m technical report, 2024. Qwen Team, Alibaba Group.

M. Zampieri, S. Malmasi, P. Nakov, S. Rosenthal, N. Farra, and R. Kumar. 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. Association for Computational Linguistics, 2019. doi: 10.18653/v1/N19-1144. URL `https://aclanthology.org/N19-1144/`.

T. Z. Zhao, E. Wallace, S. Feng, D. Klein, and S. Singh. Calibrate before use: Improving few-shot performance of language models. In M. Meila and T. Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 12697–12706. PMLR, 18–24 Jul 2021. URL `https://proceedings.mlr.press/v139/zhao21c.html`.