Masteroppgave

NTNU
Norges teknisk-naturvitenskapelige universitet
Fakultet for informasjonsteknologi og elektroteknikk
Institutt for informasjonssikkerhet og kommunikasjonsteknologi

Kevin Hjelmtveit

Moderating Respectful Conversations Using Reinforcement Q-Learning and Markov Decision Process Approaches

Considering Toxicity in Speech

Masteroppgave i Informasjonssikkerhet
Veileder: Sule Yildirim Yayilgan
Desember 2023

NTNU
Kunnskap for en bedre verden

Kevin Hjelmtveit

Moderating Respectful Conversations Using Reinforcement Q-Learning and Markov Decision Process Approaches

Considering Toxicity in Speech

Masteroppgave i Informasjonssikkerhet
Veileder: Sule Yildirim Yayilgan
Desember 2023

Norges teknisk-naturvitenskapelige universitet
Fakultet for informasjonsteknologi og elektroteknikk
Institutt for informasjonssikkerhet og kommunikasjonsteknologi

NTNU
Kunnskap for en bedre verden

Moderating Respectful Conversations Using Reinforcement Q-Learning and Markov Decision Process Approaches

Kevin Hjelmtveit

15.12.2023

Abstract

Today, a growing problem of hate speech is rampant in a world where digital communication dominates. This thesis investigates the use of reinforcement learning, especially Markov Decision Processes (MDPs) and Q-learning algorithms, to moderate hate speech and promote respectful dialogue on online platforms. The thesis explores the theoretical foundations of reinforcement learning, illustrating how agents can optimize their actions based on cumulative rewards and penalties. The research proposes hate speech moderation as an MDP, in which states are the toxicity levels in conversations, and actions are the responses to moderation. Through the Q-learning algorithm, a moderation agent is trained to determine the most effective response to toxic discourse scenarios based on cumulative rewards. The proposed system is tested on simulated conversations of varying toxicity levels. The results demonstrate the system's ability to moderate content accurately and adapt its responses dynamically. The thesis also addresses potential ethical concerns and suggests directions for future research in automated content moderation using reinforcement learning.

Sammendrag

I en tid hvor digital kommunikasjon er dominerende, står vi overfor et økende problem med hatprat. Denne oppgaven tar for seg anvendelsen av forsterkende læringsteknikker, spesielt Markov-beslutningsprosesser (MDP) og Q-læringsalgoritmer, i kampen mot hatprat på nettet og for å fremme en mer respektfull dialog. Avhandlingen dykker ned i de teoretiske fundamentene for forsterkende læring, og viser hvordan intelligente agenter kan finjustere sine handlinger gjennom kumulative belønninger og straffer. Oppgaven presenterer en ny tilnærming hvor moderering av hatprat modelleres som en MDP. Her representerer tilstandene forskjellige grader av toksisitet i samtaler, mens handlingene omfatter ulike moderingsteknikker. Ved hjelp av Q-læringsalgoritmen utvikles en agent som lærer å velge den mest effektive responsen til toksiske dialoger, basert på akkumulerte belønninger. Systemet evalueres gjennom tester på simulerte samtaler med varierende toksisitetsnivåer. Resultatene viser at det kan moderere innhold på en nøyaktig og dynamisk måte. Avhandlingen adresserer også potensielle etiske problemstillinger og skisserer fremtidige forskningsretninger innen automatisert moderering av innhold ved hjelp av forsterkende læring

Acknowledgement

I would like to express my gratitude to Professor Sule Yildirim Yayilgan for her valuable guidance during my thesis journey. Her critical thinking skills, understanding of the field, and collaboration have shaped my work. Sule's ability to dissect complex concepts and provide insightful feedback has enhanced my project. I am appreciative of her support and mentorship.

Contents

Abstract . . . . . **iii**
Sammendrag . . . . . **v**
Acknowledgement . . . . . **vii**
Contents . . . . . **ix**
Figures . . . . . **xiii**
Code Listings . . . . . **xv**
1 Introduction **1**
1.1 Problem description 1
1.2 Justification, Motivation, and Benefits 2
1.3 Research questions 3
1.3.1 Research Question 1: 3
1.3.2 Research Question 2: 3
1.3.3 Research Question 3: 3
1.3.4 Research Question 4: 3
1.4 Planned Contributions 4
1.5 Thesis Outline 4
2 Background **5**
2.1 Digital Toxicity: The Evolution of Hate and Offense 5
2.1.1 Definition of Hateful and Offensive Language 6
2.2 Hate speech Detection & Reinforcement Learning 7
2.2.1 Detection of Hate speech Online 7
2.2.2 Respectful Conversation in Online Platforms 8
2.2.3 Reinforcement Learning for Hate Speech Detection 8
2.2.4 Markov Decision Processes (MDP) 10
2.2.5 Challenges in Hate Speech Detection 12
2.3 Related work 13
2.3.1 Hate Speech & Social Impact 13
2.3.2 Online Hate Speech Detection 15
2.3.3 Reinforcement learning for Conversation Management . . . 18
2.4 Research Conclusion 20
2.4.1 Hate Speech & Social Impact 21
2.4.2 Online Hate Speech Detection 21
2.4.3 Reinforcement learning for Conversation Management . . . 21
3 Methodology **23**

3.1 System overview . . . 23
3.1.1 Software Methodology . . . 23
3.1.2 Moderator and Users . . . 24
3.2 Reinforcement Learning . . . 24
3.2.1 Markov Decision Process (MDP) Methodology . . . 24
3.2.2 Detoxify: Pre-trained Toxic Analyzer . . . 25
3.2.3 Detoxify: Implementation . . . 26
3.2.4 Q-Learning and Integration with Detoxify . . . 28
3.3 Data Preprocessing . . . 29
3.3.1 Dataset . . . 29
3.3.2 Dataset Construction . . . 29
3.3.3 Conversation Design . . . 32
3.3.4 Data Structuring, Normalization and Annotation for Analysis 32
3.3.5 Validation, Optimization, and Testing . . . 32
3.4 Limitations and Ethical Considerations . . . 33
4 Design . . . 35
4.1 System Architecture overview . . . 35
4.1.1 Moderator and Users . . . 36
4.1.2 Detoxify: implementation . . . 37
4.2 Reinforcement Learning . . . 37
4.2.1 Markov Decision Process (MDP) Framework . . . 38
4.2.2 Q-Learning Algorithm . . . 40
4.2.3 Key Components of the Q-Learning Algorithm . . . 41
4.2.4 Purpose and Functionality . . . 42
4.3 Dataset . . . 42
4.3.1 Dataset 1 - Baseline Conversational Reference . . . 42
4.3.2 Dataset 2 - Extended Conversation . . . 43
4.3.3 Dataset 3 - Extended Conversation (continue) . . . 43
4.4 Data Handling and Visualization . . . 44
5 Result . . . 45
5.1 Experiment 1: Initial Q-Learning Implementation . . . 45
5.1.1 Parameters . . . 45
5.1.2 Conversation . . . 46
5.1.3 Conclusion of Experiment 1 . . . 49
5.2 Experiment 2: Parameter Tuning for Enhanced Moderation . . . 50
5.2.1 Parameters . . . 51
5.2.2 Conversation . . . 52
5.2.3 Conclusion of Experiment 2 . . . 55
5.3 Experiment 3: Parameter Tuning for Enhanced Moderation (Continue) . . . 57
5.3.1 Parameter . . . 57
5.3.2 Conversation . . . 57
5.3.3 Conclusion of Experiment 3 . . . 59
5.4 Chapter Conclusion . . . 60

6 Discussion **63**
6.1 Reinforcement learning 63
6.1.1 Discussion: Detoxify & Markov Decision Process (MDP) . . . 63
6.1.2 Discussion: Q-learning algorithm 64
6.2 Dataset and experiment 65
6.2.1 Discussion: Experiment 1 65
6.2.2 Discussion: Experiment 2 66
6.2.3 Discussion: Experiment 3 67
7 Conclusion & Future Work **69**
Bibliography **71**

Figures

2.1 The Markov Decision Process 11

3.1 Reinforcement Learning cycle 25

4.1 System Architecture 36

5.1 Toxic score for UserA and UserB for the interaction 48
5.2 Cumulative score for UserA and UserB throughout the interaction . 48
5.3 General Toxicity Score and Q-table heatmap 49
5.4 Toxic score for UserA and UserB for the interaction 53
5.5 Cumulative score for UserA and UserB for the interaction 54
5.6 General Toxicity Score and Q-table heatmap 54
5.7 Toxic score for UserA and UserB for the interaction 59
5.8 General Toxicity Score and Q-table heatmap 59

Code Listings

3.1 Detoxify score to MDP 28

4.1 Toxic score calibration 37
4.2 Mapping toxicity score to state 38
4.3 List of possible actions 39
4.4 Transition function 39
4.5 F1-score 40
4.6 Q-learning implementation 41
4.7 Data Visualization 44

5.1 Experiment 1 Conversation 46
5.2 Experiment 1 F1-Scores 46
5.3 Experiment 2 Conversation 52
5.4 Experiment 2 F1-Scores 52
5.5 Experiment 3 Conversation 57
5.6 Experiment 3 F1-Scores 58

Chapter 1

Introduction

Advancements in technology have revolutionized communication, allowing for increased connectivity and the exchange of ideas [1], however, this has also facilitated the spread of harmful content. The digital revolution, driven by engagement-focused algorithms, can unintentionally polarize content, obscuring public and private boundaries and posing societal challenges [1–3]. The rise of hate speech and offensive language online necessitates examination and potential intervention strategies. Anonymity and the vast reach of digital platforms potentiate the spread of hate speech, manifesting as discriminatory behavior and incitement of violence against individuals or groups [4]. Originally seen as bastions of free speech and open dialogue, online platforms have increasingly become breeding grounds for hate speech. The anonymity they provide emboldens individuals to act in ways they might avoid in person. The 2018 Anti-Defamation League (ADL) report underlines the pivotal role social media plays in spreading hate speech and extremist ideologies [5, 6]. The impact of online hate speech often spills into the real world, potentially inciting violence, deepening societal rifts, and cultivating fear and intolerance. As technology continues to shape interactions, it reflects and amplifies broader social and cultural trends, presenting both challenges and unprecedented opportunities [7].

This thesis examines the relationship between language and society in the digital age. Central to this exploration is the critical issue of hate speech proliferation on online platforms, a phenomenon intensified by technological advancements. By applying reinforcement algorithms such as Markov Decision Processes (MDPs) and Q-learning algorithms, this research aims to develop innovative strategies for moderating online discourse.

1.1 Problem description

In the digital era, the proliferation of hate speech and offensive language presents a societal challenge, affecting both online and offline realms. Hate speech, characterized by communications that denigrate, discriminate, or incite violence against

individuals or groups based on attributes such as race, religion, ethnicity, gender, or sexual orientation, has been exacerbated in the digital landscape [8]. This problem description endeavors to shed light on the complex nature of hate speech, delineating its origins, various expressions, and extensive repercussions. Recent empirical research underscores the systemic nature of online hate speech, transcending isolated incidents into a widespread phenomenon. A study conducted by the Pew Research Center revealed that approximately 41% of Americans have encountered online harassment, with a disproportionate impact on racial and ethnic minorities and women [9]. Such findings amplify the critical need for effective strategies to mitigate hate speech in digital domains.

1.2 Justification, Motivation, and Benefits

This thesis focuses on the escalation of hate speech in the digital space. Communication increasingly takes place online, posing a serious threat to civil discourse and social cohesion. This thesis is motivated in part by a noticeable gap in existing studies, particularly regarding the application of advanced artificial intelligence (AI) techniques to moderate and encourage respectful online interactions. It is notable that, despite the growing research addressing online toxicity, little research has been conducted specifically on the application of reinforcement learning methodologies, including Markov Decision Processes (MDPs) and Q-learning, to online conversation moderation. This gap reflects both a challenge and an opportunity; it highlights the need for innovative approaches to address the evolving landscape of online interactions while also demonstrating AI's potential in this field. Filling this gap would provide a more efficient strategy for online platforms to moderate hate speech, thereby enhancing user safety and promoting more respectful online interactions. Reinforcement learning (RL), specifically MDPs and Q-learning, offers a dynamic and adaptive approach that is essential in the evolving landscape of digital communication. This methodology will improve accuracy in maintaining respectful conversation but also offers a scalable solution for platforms where the volume of content makes manual moderation impractical.

Furthermore, a successful implementation of this approach can significantly improve the user experience by promoting safer and more tolerant online communities. It can also assist platforms in maintaining regulatory compliance and enhancing their public image by effectively managing online toxicity. Should the approach encounter challenges or fall short of its goals, the research still holds substantial value. It will shed light on the complexities and limitations of applying advanced AI techniques in real-world scenarios, particularly in the nuanced and sensitive area of hate speech moderation.

1.3 Research questions

The research questions are a critical component of guiding the efforts and activities during the thesis, and they are part of the discussion of the chosen methodology:

1.3.1 Research Question 1:

Do online users perceive anonymity as a shield that encourages them to engage in hate speech, and how does this vary across online environments?
This research question focuses on the concept of online anonymity and its impact on hate speech. It investigates whether individuals view online anonymity as a protective shield that empowers them to express hateful views that they might otherwise withhold in offline settings. Furthermore, it explores the variations in how anonymity is perceived and utilized in online spaces.

1.3.2 Research Question 2:

How do social platforms influence the prevalence and intensity of hate speech, and what mechanisms can be identified that either foster or mitigate this effect?
This research question examines the impact of social media platforms on the nature and extent of hate speech. It seeks to understand how specific features and policies of these platforms contribute to either the escalation or reduction of hate speech. The study will explore mechanisms such as community guidelines, moderation policies, and algorithmic content promotion that may play a role in influencing the prevalence and intensity of hate speech

1.3.3 Research Question 3:

How is hate speech defined and recognized within online contexts?
This research question aims to explore the complexities of defining and identifying hate speech in the digital space. Identifying hate speech online is an increasingly challenging but crucial task, given the diversity and dynamism of online communication.

1.3.4 Research Question 4:

How can an Artificial intelligence (AI) powered system effectively identify and intervene in instances of hate speech?
By investigating how AI can accurately recognize hate speech and intervene to mitigate it, this research contributes to the development of AI interventions that promote and moderate respectful online conversations. Understanding the potential of AI in mitigating hate speech is crucial for fostering a digital environment that values responsible communication.

1.4 Planned Contributions

This thesis addresses the gap in applying advanced AI techniques, notably reinforcement learning (RL) and Markov Decision Processes (MDPs), to mitigate and maintain respectful conversation online. While existing studies focus on the detection of online toxicity, they often overlook the proactive moderation and encouragement of respectful discourse through artificial intelligence (AI) driven, rewards-based systems. This gap highlights the need to counteract the growing issue of hate speech on digital platforms and the potential of AI, specifically RL and MDPs, to shift content moderation from a reactive to a proactive practice. This approach aims to encourage a more respectful and inclusive online environment by rewarding positive interactions while punishing negativity. There are many benefits to be gained from this research, including technological advancements in AI and digital communication and societal improvements.

1.5 Thesis Outline

- Chapter 1 (Introduction) provides a problem statement, the goal of this thesis, and the research question.
- Chapter 2 (Background) describes an overview of trust within culture and society and other relevant concepts in this thesis.
- Chapter 3 (Methodology) gives a detailed description of the planned steps needed to address the problem statement. Describes the experiment design, project methodology, and result evaluation strategy the implemented programs, the experiment process, and the results obtained through the experiments, along with basic statistical data and visualized output.
- Chapter 4 (Design) Design outlines the methodology and design choices for hate speech detection, including data collection, preprocessing, feature engineering, machine learning models, and ethical considerations.
- Chapter 5 (Results) describes the implemented algorithms of the experiment process, and the results obtained through the experiments along with basic statistical data and visualized output.
- Chapter 6 (Discussion) Critically reflects on the course of the project, the obtained results, and the analysis, and discusses the approach and the alternative.
- Chapter 7 (Conclusion & Future Presents the summary of the thesis by explaining how the problem statement was addressed and what the actual outcome of the research was

Chapter 2

Background

This chapter aims to accomplish three main goals: (1) Give a reasonable definition of hate speech and the impact it has on society; (2) Present a brief overview of the implications of hate speech and its effect on society. (3) Present an overview of detection and mitigation strategies implemented on social media.

2.1 Digital Toxicity: The Evolution of Hate and Offense

Hateful and offensive language, an ongoing societal debate, has emerged as a concern, especially in its form as a broader spectrum of toxic language. Communication of this type has the potential to increase discrimination, incite violence, and undermine social harmony [10]. Throughout history, hateful speech has always existed, deeply rooted in human communication and historical prejudice. As a result of digital platforms, their characteristics and challenges have been drastically changed. Defined broadly, hate speech includes language or expressions that target individuals or groups based on attributes such as race, religion, ethnicity, gender, sexual orientation, or other identifiable characteristics [11]. It historically stems from societal biases and power imbalances, contributing to the marginalization and dehumanization of specific groups, leading to discrimination and violence [12].

In the digital age, the scope of hate speech has expanded, becoming what is now often referred to as 'toxic language'. Online platforms, with their instantaneous communication capabilities and global reach, have provided grounds for their growth. The relative anonymity of these platforms emboldens individuals to voice opinions they might self-censor in face-to-face interactions, amplifying divisive and harmful narratives [13]. While offline hate speech tends to be less anonymous and often has immediate social repercussions, the direct human interactions involved can either aggravate hostility or, in some cases, lead to dialogue and reconciliation [14].

With the advent of mass communication technologies such as the internet, radio, and television, hate speech, as a precursor to toxic language, has drastically evolved despite its historical presence, expanding its reach and sometimes being strategically used to propagate hate [15]. The consequences of hate speech during World War II, resulting in large-scale violence and systemic discrimination, serve as a stark reminder of the severe impact of this form of communication [16]. This historical perspective sets the stage for a deeper exploration of toxic language, including its detection and management, which will be addressed in the methodology chapter of this research.

2.1.1 Definition of Hateful and Offensive Language

Today's global society presents both digital and traditional challenges to hate speech. The rapid development of technology, especially the rise of online platforms, has given rise to new environments where hate speech can occur. This change calls for a deeper understanding of the various forms hate speech can take and its wider implications. Hate speech is broadly defined as any vocal, written, or behavioral communication that degrades, discriminates against, or incites animosity towards individuals or groups based on inherent characteristics like race, religion, ethnicity, gender, and disability, among others [17]. This encompasses derogatory remarks, slurs, threats, and the dissemination of malicious misinformation. Several authoritative organizations offer their perspectives on hate speech [18–21]:

- The United Nations describes it as "any form of communication... that uses pejorative or discriminatory language with reference to a person or a group based on who they are."
- The European Court of Human Rights defines it as "expressions which spread, incite, promote, or justify hatred based on intolerance."
- The U.S. Federal Communications Commission (FCC) identifies it as "speech that offends, threatens, or insults groups based on attributes such as race, religion, or ethnic origin."
- The Anti-Defamation League (ADL) characterizes it as "bigoted speech attacking a group based on attributes like race, religion, or ethnicity."

Discrimination based on race or ethnicity continues to be an integral aspect of hate speech. Due to the prevalence of digital platforms, these expressions have been exacerbated, highlighting the need for effective countermeasures [22, 23]. In addition to punitive measures against online hate speech, organizations are increasingly leveraging artificial intelligence (AI) tools to detect and mitigate subtle biases, encouraging a discourse enriched by diversity and inclusion [24]. It is important to note that while hate speech and offensive language are often used interchangeably, they differ in intent and target. Hate speech aims to dehumanize groups based on characteristics like race and religion. In contrast, offensive language involves language that may be vulgar, disrespectful, or rude but lacks a targeted, systemic intent to oppress. Generally, offensive language, while po-

tentially distasteful, is protected under the ambit of free speech, as it does not necessarily incite harm [25].

2.2 Hate speech Detection & Reinforcement Learning

As a result of the digital revolution, hate speech has become more prevalent and reaches a wider audience. It is easy for harmful content to spread across social media platforms, reaching audiences globally at unprecedented speeds. Anonymity online often encourages users to spread hatred without repercussions. A study by the Anti-Defamation League (ADL) in 2019 highlighted a significant increase in online hate speech and extremism, noting that platforms are exploited by extremists to spread their ideologies and radicalize individuals [26]. The online hate speech spectrum is broad, encompassing racist and xenophobic comments and even direct incitements to violence[26]. Individuals targeted by hate speech experience heightened stress, anxiety, and fear, which can severely diminish their quality of life. The European Commission's report in 2016 shed light on the severity of hate speech across Europe, drawing attention to its effects on human rights and the potential for it to incite real-world violence [27]. This is echoed by the Southern Poverty Law Center's annual reports, which map out the intricate challenges presented by hate speech and its links to extremist activities in the United States[28]. Through the use of advanced detection algorithms, there is an ongoing effort to mitigate the spread of hate speech online. The development of technological solutions is crucial to identifying and curtailing hateful content, which is one of the best ways to protect individuals and societies from its dangerous effects.

2.2.1 Detection of Hate speech Online

Artificial intelligence (AI), particularly machine learning, is central to detecting and mitigating online hate speech. Machine learning algorithms, from traditional methods like Random Forests to advanced deep learning models such as Recurrent Neural Networks (RNNs) and Convolutional Neural Networks (CNNs), have enhanced detection accuracy [29, 30]. Notable advancements include the use of NLP feature engineering techniques like word embeddings and TF-IDF, which help in distinguishing hate speech patterns and assessing the importance of words within broader textual contexts [31, 32]. Local linguistic patterns and sentiment analysis also play a role in identifying hate speech, with methods like N-grams revealing specific word combinations indicative of hate speech [33, 34]. Recent progress in deep learning has introduced sophisticated models like domain-specific word embeddings, autoencoders, and LSTM networks, further refining hate speech detection [35–37]. Tools like HateSonar, leveraging vector representations from techniques such as word2vec, Glove, and FastText, exemplify these advancements [38]. However, despite these technological strides, human intervention remains indispensable. Supervised learning models require accurately labeled data, and the nuanced discernment of human evaluators is crucial, particularly in complex

cases [39].

A systemic approach is necessary to combat hate speech effectively. This includes not only the deployment of advanced algorithms but also user education about the dangers of hate speech and simplified reporting processes. Integrating both technological solutions and human elements is essential to developing more effective and reliable hate speech detection and mitigation strategies.

2.2.2 Respectful Conversation in Online Platforms

The term "respectful conversation" is an objective concept, drawn from different studies and conversation analysis. The Office of Diversity and Inclusion at Ohio State University defines respectful dialogue as engaging in honest, thoughtful, and reflexive communication aimed at mutual understanding. This involves blending conversational skills with active listening and an appreciation for diverse experiences, providing a space for individuals to communicate authentically and vulnerably across differences [40]. Building on these principles, philosopher Paul Grice's Maxims provide a foundational framework for cooperative conversation. These maxims include quantity, quality, relationship, and manner, guiding participants towards effective and respectful communication [41]. Further expanding on this notion, Svennevig [42] describes the conversation as a joint activity of participatory actions that are sequentially organized and locally managed. This view underscores the cooperative and dynamic nature of the conversation, emphasizing the critical role of mutual understanding and adaptability in maintaining respectful dialogue. Napoles et al. [43] developed their ERICs (Engaging, Respectful, and/or Informative Conversations) framework for identifying high-quality online conversations. As seen, the term "respectful conversation" is not just an academic construct but an actionable guideline in digital communication. These frameworks form the bedrock of efforts to enhance the quality of digital discourse, ensuring that online platforms are spaces for diverse yet respectful exchanges.

2.2.3 Reinforcement Learning for Hate Speech Detection

Reinforcement learning (RL) represents a unique branch of machine learning, focusing on training agents to make decisions within an environment to maximize cumulative rewards. This approach, distinct from supervised and unsupervised learning, relies on a trial-and-error learning paradigm, where agents adapt their behavior based on feedback through rewards or penalties [44, 45]. In the context of online hate speech detection, RL offers innovative strategies for moderating such content. The thesis will explore key RL algorithms that contribute to identifying and mitigating hate speech.

Q-Learning

Q-learning, developed by Chris Watkins in 1989 [46], stands as a foundational model-free algorithm within the field of reinforcement learning. Characterized by its capacity to learn the value of actions in specific states without requiring a model of the environment [47], Q-learning effectively addresses challenges associated with stochastic transitions and rewards [48]. Central to Q-learning is the Q-function, which estimates the expected rewards for actions in given states. The Q-table is a key element in Q-learning, acting as a tangible representation of the Q-function. It is a matrix where rows represent different states and columns correspond to possible actions. The table stores Q-values for every state-action pair, facilitating the decision-making process for the learning agent. As the agent explores the environment, these Q-values are continually updated using the Bellman equation, as shown in equation 2.1[49]:

$$Q^{new}(S_t, A_t) \leftarrow (1-\alpha) \cdot Q(S_t, A_t) + \alpha \cdot (R_{t+1} + \gamma \cdot \max_a Q(S_{t+1}, a)) \tag{2.1}$$

Where:

- $Q^{new}(S_t, A_t)$ is the updated Q-value for the state-action pair.
- α denotes the learning rate, impacting the integration of new information.
- $Q(S_t, A_t)$ represents the current Q-value.
- R_{t+1} is the immediate reward received after action A_t.
- γ is the discount factor, reflecting the significance of future rewards.
- $\max_a Q(S_{t+1}, a)$ predicts the highest Q-value for the next state S_{t+1} across all possible actions

The Q-table, in conjunction with the Bellman equation, facilitates the dynamic updating of Q-values, blending previous knowledge with new experiences. This ensures that the algorithm progressively converges towards an optimal policy, underscoring the importance of balancing immediate rewards with the anticipated value of future actions [50]

Epsilon-Greedy Policy

In reinforcement learning (RL), the Epsilon-Greedy Policy balances exploring new actions with exploiting known ones his balance is enhanced by the epsilon (ε) parameter [44]. For instance, in the context of online hate speech detection, this policy guides RL agents in adaptively recognizing evolving linguistic patterns of hate speech. The agent alternates between exploring indicators of hate speech and leveraging established detection methodologies [51]. The ε parameter's role is key, allowing adjustment between exploration and exploitation, which is essential given the dynamic nature of online discourse [52]. This adaptability is important in the continually evolving field of online communication, where new forms of hate speech frequently emerge, demanding a detection system capable of

real-time evolution. The Epsilon-Greedy Policy serves as a strategic tool for maintaining the efficacy of hate speech detection systems in the ever-changing online landscape.

Reinforcement learning (RL) algorithms, such as Q-learning and Epsilon-Greedy policies, can be strategically applied to cultivate respectful conversation in online platforms through the principle of positive reinforcement. This approach involves designing RL systems to identify and reward user interactions that exhibit positive, supportive, and inclusive behavior. Such a methodology aligns with the principles of behavior modification, notably positive reinforcement, a concept extensively explored in the field of psychology. By reinforcing constructive and respectful user engagement, these algorithms play a pivotal role in shaping online discourse dynamics and supporting a more positive and inclusive digital communication environment. This strategy has its roots in the psychological framework of behavior modification, emphasizing the efficacy of positive reinforcement in influencing behavior [53].

2.2.4 Markov Decision Processes (MDP)

As the primary focus of this thesis, the Markov Decision Process (MDP) represents an environment where decisions are accompanied by stochastic results [54]. As part of dynamic programming and based on operations research models, MDPs are gaining increased attention, especially for reinforcement learning (RL) [54, 55]. A set of states encapsulates every possible scenario of the environment, during navigation of these states, agents can choose from a range of decisions, shown as actions seen in figure 2.1. The key property of a MDP is its deterministic-stochastic dynamics, represented by the transition model $P(s'|s,a)$. This model presents the probabilistic understanding that action a taken in state s transitions to state s' [56, 57]. Enhancing this transition is the reward function, denoted $R(s,a,s')$, which numerically appraises the immediate outcomes associated with each state-action interaction. To harmonize the interplay between immediate and future rewards, the discount factor γ, a scalar between 0 and 1, calibrates the value agents place on near-term versus extended temporal outcomes.

The process of MDP can be explained using the figure 2.1 below

- States (S): These are the nodes where decisions are made. From the image, there are three states represented by the circles labeled S_0, S_1, and S_2.
- Actions (A): These are the options or decisions available at each state. They are typically represented as arrows pointing to the states. The actions are labeled as a_0 and a_1.
- State Transitions: These are implied by the arrows that connect one state to another. These arrows are associated with probabilities indicating the likelihood of transitioning from one state to another given an action.
- Rewards: These are typically numerical values associated with transitions

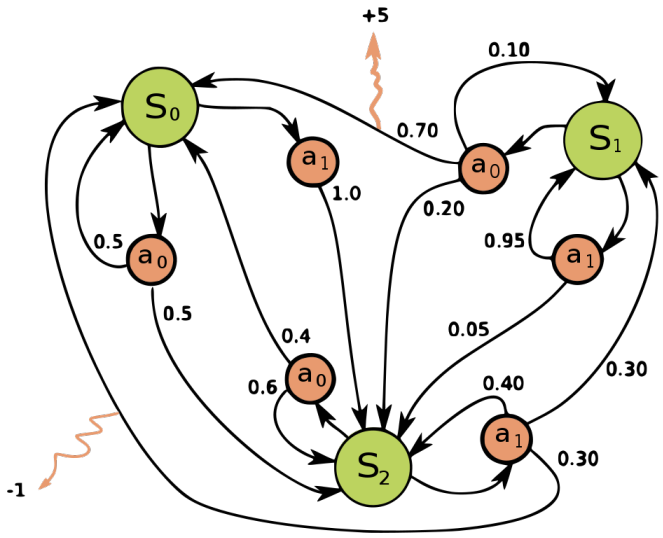


Figure 2.1: The Markov Decision Process
[56]

from one state to another due to an action taken. In the image above, these are represented as "squiggly" lines with a "-1" and "+5". In the context of MDPs, a negative reward typically represents a cost or a penalty. While, the "squiggly" line with a "+5" would indicate a reward of +5, which would be a benefit or positive outcome of taking an action from a state.

In Markov Decision Processes (MDPs), knowledge is acquired, and the most effective policy is selected based on rewards. The total expected rewards increase over time as a result of such policies. As a result of MDPs, decisions are not only guided by immediate rewards but also by potential future rewards, ensuring that actions align with the overarching strategy [58, 59]. MDPs provide a structured approach to reinforcement learning (RL), which is based on discrete decision-making sequences. In response to feedback they receive from the environment, agents can fine-tune their strategies. A variety of decision-making processes can be captured using MDPs in the RL context, according to Kaelbling et al. [60]. It is possible to use MDPs to prevent hate speech, for example, by modeling user interactions over time and adapting to changes in the dynamics of conversations as they evolve. With this flexibility, platforms can make informed decisions about content moderation and user warnings, protecting community standards and preventing escalation. MDPs are governed by the Bellman Equation, a mathematical construct establishing a recursive relationship between state value and its successors [61]. In parallel with MDP's rise in the 1990s, Q-learning emerged as a model-free gem in the reinforcement learning field [62]. This iterative approach

strives to map the optimal action-value function, represented as $Q^*(s, a)$, illuminating the anticipated returns of actions within states. The learning dynamics are captured by the update equation, which harmoniously integrates prior knowledge with newfound insights.

2.2.5 Challenges in Hate Speech Detection

Through a combination of classical approaches and new innovations, Q-learning and deep learning have produced frameworks like Deep Q Networks (DQNs). As a result of using deep neural networks, these combined systems are more capable of dealing with complex fields, including games and robotics [63].

There are many obstacles to identifying hate speech, such as the ambiguous and context-sensitive nature of language. Depending on their contextual context, words or expressions can be perceived differently, making it difficult to determine whether they are offensive. One scenario may consider a term objectionable, while another may perceive it in a completely different light [64, 65]. For machine learning models to be effective, they need to be trained on data that is representative and high quality. A model trained on biased or unrepresentative data is likely to exhibit skewed performance in real-world settings. The myriad forms hate speech takes can compound this challenge, so a comprehensive and balanced dataset is required for effective model training [66].

Online, language presents a significant challenge due to its fluidity. Hate speech is often disguised as a new term or allusion to avoid detection by those who engage in it. This dynamic environment highlights the need for ongoing updates and retraining of fixed models to keep up with the changing patterns of hate speech [67]. The process of creating strong datasets for detecting hate speech is complex and fraught with difficulties. An assessment of how effective detection algorithms perform in specific contexts involves pinpointing affected demographics, collecting unprocessed data, establishing a process for annotation, and establishing a process for annotating data. The lack of comprehensive, superior-quality datasets in this field frequently hinders progress [68]. The escalation of hate speech online has necessitated the implementation of automated detection and moderation mechanisms by various platforms. While promising, this advancement is accompanied by numerous obstacles that demand thorough research and analysis. The complexities of accurately identifying hate speech via automation are a particular point of focus within computational social science [69]. Addressing these issues highlights the complex aspects of monitoring hate speech, calling for a comprehensive, knowledgeable, and adaptive strategy to combat the issue effectively.

2.3 Related work

This section presents related research within the fields of socio-cultural aspects and the impact of hate speech, as well as the various detection and mitigation strategies utilizing artificial intelligence and reinforcement learning. In addition, investigating similar research may justify applying the methodology to answer the research questions as presented in 1.3. This section provides a review of the current state of research and discusses how this thesis might offer some insight into these fields. The relevant research ensures that the research being conducted is unique and has the potential to benefit the research community.

2.3.1 Hate Speech & Social Impact

The research by Mathew et al. [70] is about understanding the diffusion dynamics in Gab, a social networking site, and analyzing the differences in diffusion of posts generated by hateful and non-hateful users. The researchers collected a massive dataset of posts and users by following the crawling methodology in Zannettou et al. [70]. They used Gab's API to crawl the site in a snowball methodology and collected the data for the most popular user as returned by Gab's API, then collected the data for all their followers and followings. The dataset includes basic details about each user, like username, score, account creation date, all the posts of each user, and all the followers and followings for each user. The researchers used a random sample of 200 accounts per class to keep the monetary cost manageable. The research aimed to identify the diffusion characteristics of the posts made by hateful and non-hateful users and analyze the differences in the diffusion of posts generated by them. The researchers expected that the posts made by hateful users would spread further, faster, and wider. The research was conducted by collecting a massive dataset of posts and users by following the crawling methodology in [70]. The dataset includes basic details about each user like username, score, account creation date, all the posts of each user, and all the followers and followings for each user. The researchers used a random sample of 200 accounts per class to keep the monetary cost manageable. The result of the research was that the posts made by hateful users tend to spread farther, faster, and wider. These hateful users are densely connected and generate almost 1/5th of the content on Gab, despite comprising 0.3% of the users. The research was as expected, and the researchers found that the posts made by hateful users tend to spread farther, faster, and wider. The limitations of the research include the fact that the dataset only includes publicly available data posted on Gab, and the researchers could not trace the exact influence path.

The research by Zannettou et al. [71] aims to understand how memes, which are images, videos, or slogans that spread online and often evolve, are created and shared across different web communities and how they influence public opinion and culture. The paper focuses on fringe web communities, which are platforms

that allow users to post any content without censorship, such as 4chan's Politically Incorrect board (/pol/), Gab, and The Donald subreddit. These communities are known for producing and disseminating hateful and extremist memes, which can have negative social and political consequences. The research collects and analyzes a large dataset of 160 million images from 2.6 billion posts gathered from X (former Twitter), Reddit, /pol/, and Gab over 13 months. The paper uses a technique called perceptual hashing to group similar images into clusters and then annotates them using meme metadata obtained from Know Your Meme, a website that documents the origin and evolution of memes. The paper also maps the images from mainstream communities such as X and Reddit to the clusters to measure the propagation and influence of memes across the web. The paper concludes that memes are a serious phenomenon in online social media and that they can be used to sway and manipulate public opinion, as well as to promote hate and extremism. The paper calls for more research and intervention to prevent the harmful effects of memes and encourage a more respectful and constructive online discourse.

C. Yong [72] explores the inclusion of hate speech within the Free Speech Principle (FSP). The author argues that hate speech is a diverse category and identifies four main subcategories, examining each of the FSP's justifications. Ultimately, the author advocates for distinguishing between types of hate speech and differentiating coverage and protection. The FSP is a distinct principle in political morality that safeguards freedom of speech, encompassing all forms of communication, both verbal and non-verbal, aimed at conveying a message. It asserts that individuals should be shielded from viewpoint discrimination, imposing a moral constraint against violating the FSP. Consequently, restrictions on speech require a more rigorous justification than restrictions on other activities.

Howard et al. [73], discuss the moral and legal considerations that underlie the right to free speech and how they relate to hate speech. The author argues that while there are strong moral reasons to protect free speech, there are also moral reasons to restrict hate speech. The author suggests that one fruitful future line of inquiry for those sympathetic to the democratic argument is to find a persuasive answer to the question of why, if democratic citizens are committed to fundamental rights, they nevertheless have the prerogative to advocate the adoption of such hateful legislation in their public discourse. The author also discusses the limitations of their methodology and suggests that one promising methodology for exploring questions like this is to identify the considerations that serve to justify the moral right to freedom of expression in the first place and then inquire whether a controversial category of speech is sufficiently related to these considerations to qualify for protection. The author suggests that future work should focus on finding a persuasive answer to the question of why democratic citizens have the prerogative to advocate the adoption of such hateful legislation in their public discourse.

The paper by Napoles et al. [43] focuses on identifying high-quality online conversations, termed ERICs (Engaging, Respectful, and/or Informative Conversations), particularly in the context of responses to news articles and in debate forums. The objective is to develop a model capable of automatically categorizing good online conversations, driven by the challenge of sifting through the vast volumes of online discourse to find quality interactions. The results are significant, with the model achieving F1 scores of 0.73 in online news article responses and 0.91 in debate forums, demonstrating a high level of accuracy in identifying ERICs. These findings are crucial as they show the systematic and accurate identification of good conversations on online platforms is feasible. The conclusion drawn from this research underlines the feasibility and importance of curating quality conversations online, providing a foundational step for future research and development in online conversation analysis and moderation. Looking forward, the paper suggests the potential expansion of the model's applicability across various online platforms, refinement to include more nuanced criteria for good conversations, and exploration of practical applications, such as integrating social media platforms to enhance user experience and content quality.

2.3.2 Online Hate Speech Detection

The research paper by Qian et al. [74], explores the critical challenge of countering online hate speech through the use of natural language processing (NLP) techniques. The authors discern a pressing need to transcend the conventional boundary of merely detecting hate speech, advocating for a more proactive approach to discourage individuals from perpetuating hate speech in online conversations. Their proposition lies in a novel task termed "generative hate speech intervention," envisioned to automate the generation of responses that can intervene in online discussions tainted with hate speech. They introduced two robust, fully-labeled datasets harvested from the social media platforms Gab and Reddit, which capture conversation segments, hate speech labels, and intervention responses crafted by Mechanical Turk Workers. These datasets, containing 5000 conversations from Reddit and 12000 from Gab, are distinctive as they retain the conversational context and feature human-written intervention responses, which are deemed crucial for developing generative models aimed at curtailing the proliferation of hateful dialogues online. Furthermore, the paper embarks on a journey of dissecting common intervention strategies and gauging the performance of automatic response generation methods on these fresh datasets, laying down a benchmark for ensuing research in this domain. Their contributions are threefold: ushering in the generative hate speech intervention task along with the accompanying datasets laden with human-written intervention responses, encapsulating data in the guise of conversations for enriched context, and bridging a research chasm concerning hate speech on Gab and Reddit by delivering datasets that are also primed for hate speech detection tasks owing to the meticulous labeling of

posts as hate or non-hate speech by Mechanical Turk workers.

The research by Thomas Davidson et al. [67] tackles the complex issue of distinguishing hate speech from other forms of offensive language on social media platforms. The primary challenge in automatic hate speech detection, as identified by the authors, is the differentiation of hate speech from other offensive language instances. Existing lexical detection methods, which often classify all messages containing specific terms as hate speech, have been found to have low precision. Moreover, prior work using supervised learning also struggled to differentiate between the two categories effectively. To address this challenge, the authors utilized a crowd-sourced hate speech lexicon to collect tweets containing hate speech keywords. They then employed crowd-sourcing to label a sample of these tweets into three distinct categories: those containing hate speech, those with only offensive language, and those with neither. A multi-class classifier was trained to differentiate between these categories. Through a close analysis of the predictions and errors, the authors were able to ascertain when the differentiation between hate speech and other offensive languages could be reliably made and when it proved more challenging. Notably, the authors found that tweets with racist and homophobic content were more likely to be classified as hate speech, whereas sexist tweets were typically classified as merely offensive. Additionally, tweets lacking explicit hate keywords were found to be harder to classify. The paper goes beyond mere identification and looks into the nuances that differentiate hate speech from other offensive language forms, making significant strides toward enhancing the precision of automated hate speech detection systems. Through their methodical approach, the authors not only highlight the limitations of existing lexical and supervised learning methods but also present a refined classification system that can better navigate the nuanced landscape of online language. Their work underscores the need for future research to better account for context and the heterogeneity in hate speech usage, thereby contributing to the broader endeavor of creating safer online spaces.

The research by Johannes Welbl et al. [75], investigates the toxicity inherent in large language models (LLMs) and proposes mitigation strategies. The authors underscore the necessity of ensuring the safety of text generated by LMs for their real-world deployment, especially considering their remarkable fluency across various natural language processing tasks. The discourse critically assesses the prevalent practice of automatic toxicity evaluation of LMs and juxtaposes various toxicity mitigation strategies through both automatic and human evaluation. A noteworthy proposition within the paper is the Causally Fair Language (CFL) architecture, designed to detoxify pre-trained LLMs in a plug-and-play fashion. This architecture, based on a structural causal model (SCM), is lauded for its mathematical transparency and computational efficiency, setting it apart from many existing detoxification techniques. Besides, the paper also touches upon the potential of large LLMs, trained on vast text corpora, in diverse applications, includ-

ing few/zero-shot learning and code generation, while also highlighting pressing issues like distributional biases, social stereotypes, and the potential revelation of training data which necessitates address.

The research by Da Silva et al. [76]. explores the capacity of conversational artificial intelligence (AI) systems to identify offensive language, focusing on the challenges and potential solutions surrounding this issue. The authors explore how the widespread use of AI, particularly natural language processing (NLP) and machine learning (ML), has enabled the development of Chatbots and conversational AI applications. However, they also address the inherent biases in these technologies stemming from historical data used to train them, which can result in unethical behavior when processing language. The researchers present a low-level proof-of-concept to understand the challenges faced in detecting offensive language, particularly on social media platforms. They examine existing literature to explore whether chatbots and conversational AI can effectively deal with offensive language and discuss the limitations of these AI technologies in this context. The authors also mention the proactive detection capabilities of AI tools used by companies like Facebook, which had a detection rate of 97.1% for hate speech posts during the fourth quarter of 2020. They explore the possibility of extending such promising approaches to conversational AI to enhance offensive language detection. Additionally, they touch upon the repercussions of ethical ideology, social competence, and perceived human likeness on the use of offensive language in human-AI chatbot interactions. The paper further elaborates on how large language models often exhibit unethical behavior due to biases in historical data, and suggests some steps to achieve stronger results in offensive language and unethical behavior detection using Conversational AI technologies.

The research by Badjatiya et al. [77], explores the application of deep learning methodologies for identifying hate speech within tweets on X (former Twitter). This task is crucial for various applications such as the extraction of controversial events, the development of AI chatterbots, content recommendations, and sentiment analysis. The goal is to categorize a tweet as racist, sexist, or neither, which is a complex task due to the intricacies of natural language constructs. The methodology of the research included extensive experiments with multiple deep learning architectures to learn semantic word embeddings to tackle this complexity. Various classifiers like Logistic Regression, Random Forest, SVMs, Gradient Boosted Decision Trees (GBDTs), and Deep Neural Networks (DNNs) were experimented with. The feature spaces for these classifiers were defined by task-specific embeddings learned using three deep learning architectures: FastText, Convolutional Neural Networks (CNNs), and Long Short-Term Memory Networks (LSTMs). These methods were compared with baseline approaches, which used feature spaces comprising char n-grams, TF-IDF vectors, and Bag of Words vectors (BoWV). The main contributions of the research are the investigation of deep learning methods for hate speech detection, the exploration of various tweet se-

mantic embeddings, and significant improvements over state-of-the-art methods. The experiments were conducted on a benchmark dataset of 16,000 annotated tweets, which were labeled as sexist, racist, or neither. The dataset was utilized to train and fine-tune the neural network architectures, and once the networks were trained, new tweets were tested against the network to classify them as racist, sexist, or neither. The study found that deep learning methods significantly outperformed the state-of-the-art char/word n-gram methods by a notable margin, enhancing the capabilities of automated systems in identifying hate speech on X (former Twitter).

The Jigsaw research [78] and dataset from the "Toxic Comment Classification Challenge" represent a significant stride in the field of natural language processing (NLP), particularly in addressing the pervasive issue of online toxicity. This dataset, derived from Wikipedia's talk page edits, encompasses a wide array of comments annotated for various forms of toxic behavior, including general toxicity, severe toxicity, obscenity, threats, insults, and identity hate. The multi-faceted nature of this annotation allows for a comprehensive approach to understanding and categorizing online harmful speech. The primary objective of the challenge was to develop machine learning models capable of accurately identifying and classifying different types of toxic comments. This initiative was driven by the need for effective moderation tools on online platforms, where harmful content can have significant negative impacts on users and communities. By providing a rich, annotated dataset, the challenge aimed to foster advancements in automated content moderation technologies. The results of the challenge demonstrated considerable success in using machine learning to identify toxic content. Many models achieved high levels of accuracy, showcasing the potential of AI in aiding content moderation. However, the challenge also highlighted the complexity of the task, particularly in dealing with subtleties of language and context, as well as the risks of biases in the models.

2.3.3 Reinforcement learning for Conversation Management

The research by Seyed Sajad Mousavi et al. [79], is an in-depth review of the advances in deep reinforcement learning (DRL). The authors explore how deep learning, known for its ability to model high-level abstractions in data through deep architectures, can be combined with reinforcement learning methods. The fusion of these two domains aims to learn useful representations of problems, thus aiding in overcoming challenges related to perception and environmental representation, which are crucial for an agent to select optimal actions. The paper discusses the fundamental challenges in reinforcement learning, particularly the necessity for an agent to have a good representation of its environment before deciding on an action. The document also explores how deep learning has not only gained traction in academia but has been effectively utilized in industry products by tech giants like Google, Apple, and Microsoft, among others. The authors aim

to outline and critically review all significant research done to date in the context of combining reinforcement learning algorithms and deep learning methods. This research provides knowledge and insights into the capabilities of DRL that are key for developing systems for managing conversations.

The research by Jiwei Li et al. [80], explores utilizing deep reinforcement learning (DRL) for enhancing dialogue generation in conversational agents. The primary concern addressed is the shortsighted nature of recent neural models that predict utterances in isolation, disregarding their potential impact on subsequent conversational flow. The authors propose a model that simulates dialogues between two virtual agents, employing policy gradient methods to reward dialogues that exhibit three pivotal conversational properties: informativity (non-repetitive turns), coherence, and ease of answering (related to forward-looking function). The research underlines the critical role of modeling the future direction of dialogue to generate more engaging and coherent conversations, a necessity that drove traditional Natural Language Processing (NLP) models toward reinforcement learning. The DRL model introduced in the paper aims to predict future rewards in chatbot dialogues, thus facilitating more interactive and sustained conversations. This model is evaluated based on diversity, length, and human judgments, showing that the proposed algorithm generates more interactive responses and manages to foster a more sustained conversation in a dialogue simulation. The authors claim that this work marks a step towards developing a neural conversational model focused on the long-term success of dialogues.

This research by John Schulman [81] proposes a new family of policy gradient methods for reinforcement learning called Proximal Policy Optimization (PPO). The objective of the research is to improve the current state of reinforcement learning by introducing an algorithm that attains the data efficiency and reliable performance of Trust Region Policy Optimization (TRPO) while using only first-order optimization. The research was conducted by alternating between sampling data from the policy and performing several epochs of optimization on the sampled data. The experiments tested PPO on a collection of benchmark tasks, including simulated robotic locomotion and Atari game playing, and showed that PPO outperforms other online policy gradient methods and overall strikes a favorable balance between sample complexity, simplicity, and wall-time. The research was conducted using a fully connected Multi-Layer Perceptron (MLP) with two hidden layers of 64 units and tanh nonlinearities to represent the policy in the PPO algorithm. The MLP outputs the mean of a Gaussian distribution with variable standard deviations, following [82]. The policy and value functions do not share parameters, and an entropy bonus is not used. The sample data used in the experiments were from the MuJoCo environments and the Arcade Learning Environment benchmark. The results of the research showed that PPO performs better than other online policy gradient methods and strikes a favorable balance between sample complexity, simplicity, and wall time. The research also found that the

version of the surrogate objective with clipped probability ratios performed best. The limitations of the research include the need for improvement in developing a method that is scalable, data-efficient, and robust.

This research paper introduces a groundbreaking policy gradient method tailored for robust reinforcement learning in the presence of a model mismatch. The primary goal is to develop a method capable of training policies that remain robust even when faced with discrepancies between the simulation environment and the real-world setting. This paper primarily focuses on the R-contamination uncertainty set model and presents a robust policy gradient approach that offers provable assurances regarding global optimality and robustness. Additionally, it extends this methodology to the broader model-free context and formulates a robust actor-critic method with a differentiable parametric policy class and value function. To achieve this, the researchers devised a robust policy (sub-)gradient applicable to any differentiable parametric policy class. Importantly, the proposed robust policy gradient method exhibits asymptotic convergence to the global optimum when utilizing direct policy parameterization. The authors delve further into characterizing its asymptotic convergence and sample complexity in the tabular setting. Empirical simulations are provided to validate the effectiveness and robustness of the proposed techniques. The research quantifies the complexity of achieving an epsilon-global optimum in the robust policy gradient method, demonstrating it to be $\mathcal{O}(\epsilon^{-3})$.

These related works provide a diverse exploration into the realms of hate speech detection and the management of online conversations, with a specific focus on the applications of reinforcement learning (RL). They offer insights into the existing methodologies and the challenges faced in both identifying hate speech and enhancing the quality of digital discourse through reinforcement learning strategies. Building upon this foundation, this thesis aims to bridge a gap by exploring how an RL-based system, integrated with Markov Decision Processes (MDP) and Q-learning, can effectively promote and maintain positive online interactions which often has been neglected in current hate speech detection methodologies. In contrast to traditional reactive systems, this innovative approach leverages a cumulative reward curve, using it strategically to actively encourage respectful communication and discourage negative discourse.

2.4 Research Conclusion

The related research gives a good overview of the research within the different topics related to this thesis. Even though most of the related work is generalized and not necessarily pinpointed to Scandinavian culture, it does give a good overview.

2.4.1 Hate Speech & Social Impact

Hate speech has a powerful influence on online communities, and this research reveals how it influences social dynamics. Through memes and other digital artifacts, hateful content can shape public opinion and culture. Hate speech in digital domains must be understood and mitigated to prevent further harm. As a result, more comprehensive data collection and analysis methods are needed to accurately trace hate speech's influence paths. As a result of this research, future investigations will explore more effective strategies to combat hate speech online while balancing the complicated balance between free speech and hate speech.

2.4.2 Online Hate Speech Detection

The research presented in this section highlights the need for a dynamic and multifaceted approach to detecting and counteracting online hate speech. These studies show how hate speech detection has evolved from simple analysis to deep learning and generative interventions. In the future, it may be able to detect hate speech in more nuanced and contextual ways due to the emergence of new datasets and refined classification systems, as well as the inherent biases of large language models. In the rapidly evolving digital communication landscape, this section emphasizes the importance of continuous technological innovation and ethical considerations in creating safer digital spaces.

2.4.3 Reinforcement learning for Conversation Management

The research explored in this section shows the innovative application of deep reinforcement learning to enhance conversation management and dialogue generation. By combining deep learning and reinforcement learning, conversational agents can now be reactive, predictive, and contextually aware. With advances in policy optimization algorithms and the exploration of robust reinforcement learning methods, conversational systems can be more resilient and effective. Research indicates deep reinforcement learning is key to managing online discourse, especially in mitigating hate speech and maintaining respectful and meaningful interactions. Using reinforcement learning to navigate the complexities of online conversations and their societal influences is one of the areas that this research explores further.

Further, the next chapter lays out how these challenges are addressed.

Chapter 3

Methodology

The purpose of this chapter is to provide an overview of the approach taken in creating a system that can moderate conversations by reducing hate speech.

3.1 System overview

The goal is to create a system that oversees and moderates conversations, effectively reducing toxic speech and promoting a more respectful communication environment. This system is designed to be robust, scalable, and adaptable to different types of platforms, ranging from forums to social media. The approach integrates advanced computational techniques with innovative software solutions to address the challenges of moderating dynamic and complex online interactions.

3.1.1 Software Methodology

Central to our system's development is the Python programming language, chosen for its flexibility and extensive support in data science. Python offers a multitude of libraries and tools, enabling to build a powerful yet maintainable system. The full code for this thesis can be found on github [83]. The key Python libraries utilized are:

- **Numpy**: It is necessary to use Numpy for this project, it is essential to provide robust data structures but also a comprehensive set of mathematical functions.
- **JSON**: The library manages JSON data, a format important for lightweight data exchange. In the architecture, JSON is used for integral for efficient data transmission and storage.
- **Plotly**: This module uses Plotly, capable of creating interactive and visually graphical representations
- **Detoxify**: Detoxify assists in identifying and filtering toxic or inappropriate content within text data [84]. The application is essential in ensuring the integrity and appropriateness of user-generated content within the system,

protecting against potential misuse or harmful interactions.

These libraries have been selected for their synergistic potential in creating a robust and efficient system unique to our needs.

3.1.2 Moderator and Users

To ensure the system functions can oversee the conversation, a moderator will be implemented. A moderator's job is to track state, toxic score, and cumulative reward after each conversation. These metrics are important and are used further in data analysis. Further, two users will be implemented (UserA and UserB) which will simulate a conversation between two users. The conversation will be in a controlled environment with a predefined script.

The goal is to have a respectful conversation with minimum levels of toxicity. The term "respectful" is subjective, but within the context of this thesis, it would be in line with Napoles et al. [43]'s term ERICs (Engaging, Respectful, and/or Informative Conversations), which would fall into the 30% or lower on the detoxify score, which should put it in the "normal" state.

3.2 Reinforcement Learning

Reinforcement learning (RL) is a type of machine learning where an agent learns to make decisions by performing actions in an environment and receiving rewards or penalties in return. The goal is to develop a policy that maximizes cumulative rewards over time. A key component of the reinforcement learning framework is the Markov Decision Process (MDP), which models decision-making in environments where outcomes are partly random and partly controlled by the decision maker[44]. The Detoxify library, a pre-trained model for identifying toxicity in online interactions, is integrated into this framework to define the states of the MDP. This allows the RL agent to discern various levels of toxicity within the user-generated content, thereby facilitating a more nuanced moderation process [84]. Figure 3.1 shows the cycle of reinforcement learning, where the agent does some type of action that would affect the environment that influences the state and reward.

3.2.1 Markov Decision Process (MDP) Methodology

In our system, the states within the MDP framework are defined based on the toxic scores generated by Detoxify.

- **Environment:** The setting where the agent operates, which, in this case, is the platform for discourse.
- **Agent:** The learner or decision-maker interacts with the environment, in this case, two agents simulating two users.

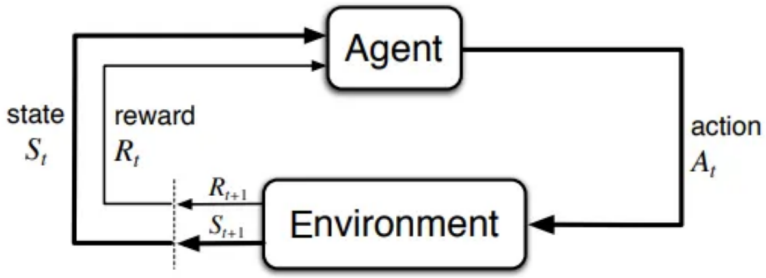


Figure 3.1: Reinforcement Learning cycle
[85]

- **States (S):** are all possible situations that the agent can encounter during the conversation. The states are the basis for the MDP. There are three states: normal, medium, and toxic.
- **Actions (A):** These are all possible moves that the moderator can take based on the current state of the conversation. The actions could range from flagging a message, ignoring it, or categorizing it under different levels of toxicity.
- **Transitions (T):** It is the transitions between states that reflect the evolving nature of discourse due to moderation actions
- **Rewards (R):** Feedback from the environment is based on the actions taken by the moderator.
- **Policy:** An agent's strategy for mapping states to actions to maximize long-term rewards
- **Q-function:** Indicator of expected future rewards for a particular action.

The agent's training process is iterative, guided by the MDP, where decisions are based on the current state and expected outcomes. To effectively moderate online platforms, develop effective reward systems, and maintain a balance between active moderation and freedom of expression, this structured decision-making process is essential. The agent learns by interacting with the environment, receiving rewards for actions taken, and updating its policy towards more rewarding actions. This learning process is iterative and can be computationally intensive, often requiring numerous episodes of interaction before a robust policy emerges. The MDP provides the formal framework for defining the interactions between the agent and the environment.

3.2.2 Detoxify: Pre-trained Toxic Analyzer

Chosen for our conversation moderation system, Detoxify is pivotal in detecting and quantifying online toxicity. It adeptly processes large datasets and learns from evolving data, outperforming traditional rule-based and lexicon-based methods that often falter in adapting to the dynamic online language landscape. A key

functionality of Detoxify is its ability to assign a toxicity score to each comment, providing a quantifiable measure of potential toxicity. These scores are integral to our approach, as they feed into a Markov Decision Process (MDP) that guides our system's response strategies. This use of advanced natural language processing (NLP) techniques for scoring, as emphasized in studies such as Schmidt et al. [86], enables a nuanced understanding of language nuances. The combination of Detoxify's efficient toxicity quantification and its integration into the MDP framework makes it a strategically chosen, effective tool for our conversation moderation system. Other benefits of using Detoxify are:

- **Time Efficiency**: By leveraging Detoxify, a model that has already been trained on extensive datasets, it significantly saves time that would otherwise be spent on the labor-intensive process of training a model from scratch. Detoxify's readiness for immediate deployment accelerates our project timeline, allowing us to focus on other critical aspects of the system.
- **Proven Accuracy and Reliability**: Detoxify's training on diverse data ensures its capability to accurately identify a wide range of toxic speech. This high level of precision is essential for the effective moderation of conversations in our system.
- **Focus on Reinforcement Learning**: The core of our project is the development of a respectful conversational moderation system using a reinforcement learning model. By employing Detoxify for toxicity classification, can direct our resources toward enhancing this aspect of the system, concentrating on innovative techniques in conversation moderation.
- **Resource Optimization**: Adopting Detoxify allows for the optimization of resources, as training a comparable model would require significant computational power and time, which may not be feasible within the scope of our project.
- **Scalability and Flexibility**: Despite being a pre-trained model, Detoxify offers adaptability. It is possible to fine-tune it to meet our project's specific needs, ensuring it effectively addresses the nuances of our system.
- **Foundation for Markov Decision Process (MDP) State**: A critical aspect of our methodology is the application of a Markov Decision Process (MDP) in our reinforcement learning model. The toxic scores provided by Detoxify serve as the foundational basis for defining the states within this MDP framework. By quantifying the level of toxicity in a conversation, these scores enable our system to make informed decisions on how to moderate the dialogue effectively. This integration is essential for creating a dynamic and responsive moderation system that adapts to the evolving nature of online conversations.

3.2.3 Detoxify: Implementation

Detoxify [84], a pre-trained model, is important in our moderation domain for evaluating user message toxicity, aiding as the basis of states within our Markov

Decision Process (MDP) model, crucial for managing toxic interactions. It utilizes deep learning, based on the reputable Hugging Face Transformers library, to categorize toxic content. The original model, trained on the Jigsaw Toxic Comment Classification dataset, is adept at identifying six types of toxicity [78, 87]:

1. Toxic: language that is rude, disrespectful, or unreasonable in a way that is likely to make people leave a conversation. This includes aggressive, hostile, or unnecessarily negative comments.
2. Severely Toxic: Language that is extremely offensive or aggressive. This might include stronger forms of hate speech, threats of harm, or very explicit and degrading content.
3. Obscene: Content that is offensive, rude, or vulgar, often involving inappropriate sexual references or swear words.
4. Threat: the language that conveys a threat of harm to others. This could be physical, psychological, or emotional harm
5. Insult: Comments that are demeaning, disrespectful, or abusive towards an individual or a group. This can include name-calling or derogatory remarks.
6. Identity: hate speech that attacks or demeans a person or group based on aspects of their identity, such as race, ethnic origin, religion, gender, sexual orientation, disability, or other similar characteristics.

For this thesis, Detoxify has been mapped into three categories: toxic, medium, and normal. The categories normal, medium, and toxic have been structured to align with the insights from the ADL [88] and Bahador's research [89]. The 'toxic' category, consists of the six forms of extreme negativity, reflecting the severity highlighted in these studies, and is identified by a high Detoxify score. The 'medium' category, captures content with moderate negativity and in the gray zone, reflecting Bahador's findings on the varied intensities of hate speech. The 'normal' category, is for low-scoring content, in line with the ADL's perspective on encouraging healthy online interactions. The three categories also simplify conversation moderation and enhance both research and practical application. With this simplification, a Markov Decision Process (MDP) model can make more efficient decisions, ensuring a straightforward classification process. Also, it avoids excessive flagging and missing critical toxic instances by striking an excellent balance between sensitivity and specificity. As a result of reducing the original six categories into three, issues of ambiguity and overlap are resolved, making the moderation process more transparent and streamlined.

- Toxic: This category corresponds to high scores in Detoxify's 'toxic',' severely toxic', 'obscene', 'threat', 'insult', or 'identity hate' categories. The threshold has been set to score above 0.7 out of 1 to classify a comment as toxic.
- Medium: This is used for texts that have moderate scores in Detoxify's categories, indicating some level of negativity or disrespect, but not excessively so. Scores between 0.3 and 0.7 might fall into this category.
- Normal: Texts that score low across all Detoxify categories (e.g., below 0.3) can be considered normal, indicating a lack of toxicity and potentially re-

spectful conversation.

In the MDP, the states have been defined based on the categories (toxic, medium, and normal) of the conversation at each step. Detoxify's scores provide a quantifiable measure of the conversation's quality, helping to determine the current state of the MDP. The toxic scoring in Detoxify operates by analyzing text input through its "predict" method, which returns a dictionary containing the toxicity type and its corresponding score [78]. These scores are instrumental in ascertaining the toxicity level of a message, which subsequently influences the moderation actions to be adopted. This functionality not only makes real-time moderation feasible but also efficient. Detoxify is employed to analyze the toxicity of user messages, generating a toxicity score that is then used to determine the state of the MDP. This is seen in the code 3.1 below:

```
result = Detoxify('original').predict(user_input)
toxic_score = result['toxicity']
state = int(toxic_score * 100)
```

Code listing 3.1: Detoxify score to MDP

Although Detoxify is an effective tool for detecting online toxicity, it presents several challenges. One of its primary limitations is the difficulty in understanding context, such as sarcasm, irony, or cultural nuances, resulting in inaccurate classifications. For Detoxify to detect toxicities across cultures, languages, and dialects, its training data must be free of bias and diverse. Detoxify also runs the risk of overgeneralizing, which could lead to undue censorship or suppression of valid expression when it improperly flags non-toxic content as toxic, especially in complex discussions.

3.2.4 Q-Learning and Integration with Detoxify

The Q-learning algorithm is utilized to learn a policy that guides moderation's actions. Through this value-based reinforcement learning (RL) method, the moderator seeks to maximize the cumulative rewards over time by making informed decisions based on the current state of the system, which in this case is defined by the toxicity scores obtained from processing user messages with Detoxify [87]. In the process of learning, the epsilon-greedy policy plays an important role in action selection. The idea behind this policy is to balance between exploration (trying out new actions) and exploitation (sticking to the currently known best action) [52]. The parameter ϵ (a value between 0 and 1) determines the probability of choosing exploration over exploitation. A higher ϵ value encourages more exploration, while a lower value leans towards exploitation. Specifically, with probability $1-\epsilon$, the policy chooses the action that has the maximum current Q-value (exploitation), and with probability ϵ, it selects an action randomly (exploration) [52].

This balance is key as it helps in discovering potentially better actions (exploration) while still making good use of current knowledge to obtain rewards

(exploitation). The epsilon-greedy policy has been widely used in various reinforcement learning applications due to its simplicity and effectiveness in balancing the trade-off between exploration and exploitation [52].

3.3 Data Preprocessing

Although Detoxify handles much of the text preprocessing internally, there are still additional steps required to align the data with the goal of the thesis.

3.3.1 Dataset

The dataset is engineered to observe the dynamics of conversations between the two simulated agents, referred to as "UserA" and "UserB". The reason for the controlled environment is to enable a detailed analysis of the progression of conversational toxicity over time. For instance, the dataset presented by Qian et al. [74] does use both comments and conversations for their research but lacks the conversational length and nuance for our experiment. While the Jigsaw dataset [78] only used comments to classify the toxicity level, for our research, the datasets presented only tracked hateful comments and limited conversation, which does not represent a full conversation between two individuals. Another important point is the conversational length and the variation of language, which is key for our research for tracking respectful conversation, meaning the conversation will be measured in all aspects of the toxic score.

This progression with the cumulative curve of conversation toxicity is an important element, as it plays a role in the application of the reinforcement learning model. The model is designed to appropriately reward or penalize users based on their assessed toxicity scores. By having a controlled environment for the dataset, it can be ensured that the data remains consistent and reliable. Such consistency is vital for an accurate analysis of how conversational dynamics influence the performance and decision-making process of the learning algorithm. In addition, the controlled dataset allows for the isolation and examination of specific factors that contribute to the escalation or de-escalation of toxicity. Understanding different moderation strategies and their application in real-world situations is important to understanding their effectiveness.

3.3.2 Dataset Construction

This thesis aims to explore the variations of human conversation, from considerate and respectful exchanges to confrontational and offensive ones. The goal is to maintain a respectful conversation by tracking the cumulative reward. To facilitate this analysis, three distinct datasets have been created. The first dataset forms the baseline, featuring a typical conversation that begins politely but gradually

turns into more contentious territory. This dataset serves as the foundational reference where the conversation is short in length, illustrating the initial shift from a respectful to a mildly disagreeable tone and as a comparison for long-form conversation.

Building upon this, the second dataset further explores the complexities of dialogue. It mirrors the conversation structure of the first dataset but extends the interaction, diving deeper into the 'grey zone' where the conversation shifts between respectful and toxic remarks. This dataset provides a longer context for understanding how dialogues can evolve and become more charged.

The third dataset advances this analysis by revisiting the same conversation structure under a more stringent and proactive moderation lens. It repeats the dialogue of the second dataset but with increased moderator intervention, flagging more comments for review. This dataset offers a perspective on how different moderation approaches can influence the trajectory and tone of online discussions, particularly in preventing or addressing disrespectful or hateful exchanges. Together, these datasets offer a comprehensive view of conversational dynamics, from their inception as respectful exchanges to their potential escalation into contentious or offensive territories, and the role of moderation in guiding these interactions.

Dataset 1: Baseline Conversational Reference

Throughout the conversation, a virtual moderator assesses each comment, marking actions as 'Ignore' and providing a 'State' (normal, medium, hateful) and a 'Score' that quantifies the level of toxicity or contentiousness. This scoring system, along with a cumulative metric (cuma), tracks the evolution of the conversation's tone. Initially, both users exchange opinions in a manner deemed 'normal' by the moderator, with a cumulative positive score. As the conversation progresses, the tone shifts slightly—UserA's remarks are marked as 'medium' due to a slight increase in confrontational language, while UserB's comments escalate to 'hateful', indicating a significant shift in tone and content. The conversation ends with UserA choosing to exit, highlighting a common outcome in online discussions when they turn contentious. This dataset serves as a crucial reference for understanding how conversations evolve from respectful to confrontational, and how users' responses contribute to this trajectory. The moderator's scoring system provides a quantifiable measure of the conversation's tone, offering a structured way to assess interaction dynamics.

Dataset 2: Extended Conversation

Dataset 2 expands upon the initial conversation from Dataset 1, expanding into a longer and more complex exchange between UserA and UserB. This conversation starts with a discussion about a sports game, but quickly escalates into a heated

debate, showcasing the shift from neutral to more contentious dialogue. Throughout this conversation, the moderator continues to assess each comment, with actions ranging from 'Ignore' to 'Flag for review' and 'Block or Delete'. The 'State' of the comments varies from 'normal' to 'medium' and 'hateful', with the associated 'Score' and 'cuma' (cumulative metric) reflecting these changes. This dataset illustrates how a seemingly benign discussion can escalate into a more aggressive exchange, with UserB's comments becoming increasingly hostile, eventually leading to an apology and a return to a friendly tone.

The interaction in this dataset is more dynamic compared to Dataset 1, with a significant fluctuation in the tone and content of the conversation. The moderator's actions and scoring system provide valuable insights into how the conversation's trajectory can shift and how interventions (like flagging or blocking) might be necessary in real-time moderation scenarios. The final reconciliation between UserA and UserB, with an agreement to meet for pizza, highlights the possibility of resolution even after a contentious exchange, providing a more comprehensive understanding of the complexities of online conversations.

Dataset 3: Extended Conversation (continue)

Dataset 3 represents the advanced stage of our conversational analysis, reiterating the dialogue from Dataset 2 but with a significant shift in moderator interventions. This dataset shows a similar conversation about a sports game, but the moderator's actions are more proactive, with many comments being flagged for review, even when the state remains 'normal'. This increased moderation intensity signifies an enhanced focus on potential escalations in conversation tone. The conversation begins with UserA and UserB discussing the game, and as the dialogue progresses, it becomes more heated. Despite the similar trajectory of the conversation to Dataset 2, the moderator's increased vigilance in flagging comments for review—even those with a 'normal' state—indicates a more cautious approach to moderating online discussions. UserB's comments, which quickly turn offensive, are initially ignored by the moderator, but later interventions become more stringent, with actions like 'Block or Delete'. The conversation ends with an apology and reconciliation, similar to dataset 2, but the journey to this resolution is marked by a heightened awareness and response from the moderator.

Dataset 3 provides insights into how different moderation strategies might influence the course of online conversations. The frequent flagging for review suggests a preventive approach, aiming to curb the escalation before it reaches a point of overt hostility. This dataset underscores the balance between allowing free-flowing conversation and maintaining a respectful, non-toxic environment in online interactions.

3.3.3 Conversation Design

The goal of the preprocessing involved designing a scripted conversation that transitions through various interaction states. The script was crafted to simulate a natural conversation while ensuring it traversed from a respectful to a disrespectful conversation and back. The intention was to create a controlled yet realistic representation of dialogues, capturing the nuances and shifts in tone and content. Each segment of the conversation was deliberately assigned an interaction state (normal, medium, toxic) to reflect the intended tone and content. This artificial state assignment was critical for creating a dynamic conversation flow that could serve as a model for our analysis. The assignment also included predefined moderator actions and scores to simulate a realistic moderation scenario.

3.3.4 Data Structuring, Normalization and Annotation for Analysis

To ensure uniformity in format and language, the conversation script was normalized. In addition, detailed annotations were added, including emotion, intended sarcasm, and context. These annotations enhance the script's nuance, making conversational shifts more understandable. After the structured script was formatted, it was aligned with our analysis methodology. The conversation was segmented into analyzable units and each state transition was clearly defined. This enabled us to examine how dialogues evolve and how different conversational elements influence the interaction as a whole.

3.3.5 Validation, Optimization, and Testing

Before finalizing the script for analysis, it underwent a validation process. This involved testing the script with a small group to ensure that the transitions between different states felt natural and realistic. Feedback from this testing phase was used to refine the conversation script further. These preprocessing steps ensured that our scripted conversation accurately represented a range of interpersonal dynamics and was aptly prepared for a detailed analysis of conversational shifts and moderation strategies.

The cumulative reward, toxic score, and general toxic score are pivotal metrics for assessing the moderation system's effectiveness. These metrics are tracked over time, providing a comprehensive understanding of the system's performance on both individual and collective levels. The cumulative reward is a measure of the moderation system's success in managing toxic interactions over time. A positive trend in cumulative rewards indicates a successful moderation policy, while a negative trend may signal a need for model refinement or policy adjustment.

The F1-score combines precision and memory, ensuring that hate speech is accurately identified and captured. It is important to have this balance, especially in datasets where hate speech may not be as prevalent. F1-score can improve the

user experience and credibility of the system, as it effectively detects toxic content while minimizing false negatives. Furthermore, it assists in system optimization, and guiding improvements. For the result, a high F1-score signifies the system's ability to promote respectful communication in online environments, maintaining a balance between accuracy and fairness in content moderation. To maintain a respectful and inclusive online discourse, this metric is important.

3.4 Limitations and Ethical Considerations

A significant limitation is the dependency on pre-trained models like Detoxify, which might harbor biases or fall short in adapting to the unique contexts of different online communities [90]. The accuracy of these models, especially in real-time scenarios on high-traffic platforms, further compounds the challenges. Customized and scalability are other limitations the system's rigidity could hinder adaptation to various moderation policies, and its scalability may be tested with a surge in interactions.

On the ethical front, artificial intelligence (AI) moderation systems could mirror or intensify societal biases ingrained in the training data, unfairly targeting certain user groups. Privacy is another concern, as handling sensitive or personal information in user messages without adequate safeguards could violate privacy norms. The issues of transparency and accountability are intertwined; providing clear rationales for moderation decisions is crucial for user trust, yet pinpointing accountability for harmful decisions is a complex endeavor. The balance between automated and human oversight is a delicate one; over-reliance on AI could overlook nuanced issues that a human moderator might catch, yet human oversight comes with its own set of challenges [90].

Chapter 4

Design

This chapter describes the design of the proposed system aimed at moderating online discourse by identifying and mitigating toxic speech. The design is established in the methodologies outlined in the preceding chapter, encompassing the Markov Decision Process (MDP) framework, the Q-learning algorithm, and the Detoxify model for toxic comment classification.

4.1 System Architecture overview

The system's framework is structured around three main components seen in figure 4.1: the user input, the Markov Decision Process (MDP) informed by Detoxify's toxicity evaluation, and the subsequent actions guided by a reinforcement learning (RL) approach. At the core of this architecture is the moderation mechanism which integrates user input with the Detoxify model to categorize the conversational state within the MDP. Based on this state assessment, the system, acting as a moderator, determines the appropriate action to take, learning and adapting over time through a Q-learning algorithm that assigns rewards or penalties to refine future interactions. The flowchart in figure 4.1 can be explained as such:

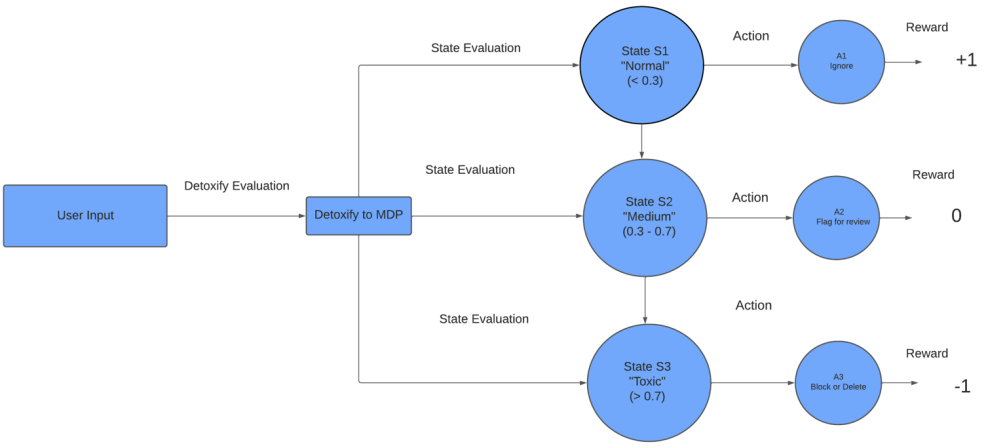


Figure 4.1: System Architecture

- User Input is received and sent to the Detoxify evaluation.
- Detoxify processes the input and assigns a toxicity score, which is then translated into an MDP state. Based on the score, the input is categorized into one of three states for further evaluation:
  - State S1 "Normal" (< 0.3): Indicating low or non-toxic content.
  - State S2 "Medium" (0.3 - 0.7): Denoting medium toxicity.
  - State S3 "Toxic" (> 0.7): Signifying high toxicity.
- An action (A1, A2, or A3) is taken corresponding to the evaluated state.Each action results in a reward:
  - +1 Reward: For actions taken in State S1.
  - +0 Reward: For actions taken in State S2.
  - -1 Reward: For actions taken in State S3, possibly

4.1.1 Moderator and Users

The moderator for the system oversees interactions between two agents, simulating two users. The objective is to moderate the conversation for a respectful conversation and interfere if the conversation goes over a certain threshold. Depending on the level of threshold, which is determined by the toxic score from our detoxify and the state of MDP applications, the moderator has the power to perform certain actions. The two users of the system (UserA and UserB) are being tracked by the moderating system to ensure which state, toxic level, and cumulative scores are present. In practice, this will not be visible to the individuals, but for research purposes, it is. The goal is to reward and punish conversations that are above a certain threshold.

4.1.2 Detoxify: implementation

As discussed in the methodology chapter 3, Detoxify has been mapped into three categories, where some categories have been merged into one from the original category:

- Toxic: This category corresponds to high scores in Detoxify's 'toxic', 'severely toxic', 'obscene', 'threat', 'insult', or 'identity hate' categories. The threshold has been set to score above 0.7 out of 1 to classify a comment as toxic.
- Medium: This is used for texts that have moderate scores in Detoxify's categories, indicating some level of negativity or disrespect, but not excessively so. Scores between 0.3 and 0.7 might fall into this category.
- Normal: Texts that score low across all Detoxify categories (e.g., below 0.3) can be considered normal, indicating a lack of toxicity and potentially respectful conversation.

The purpose of merging multiple categories into broader ones simplifies the analysis process, especially when dealing with larger datasets. This simplification also leads to operational efficiency, which is crucial in real-world applications where quick content moderation is necessary. Broader categories provide clearer communication to users or moderators, avoiding confusion from too many closely related but distinct categories. Additionally, this approach allows for more flexibility in handling language that falls into gray areas, acknowledging the complexity of speech. By adjusting the percentage score needed to classify a comment as toxic, the system's sensitivity can be calibrated according to the platform's needs, allowing for a tailored approach to content moderation. This calibration process is detailed in code 4.1 below where the criteria for categorizing comments into 'normal,' 'medium,' or 'toxic' are defined based on their toxic score.

```
# Percentage of each category
  if toxic_score < 0.3:
            state_name = "normal"
            reward = 1
       elif 0.3 <= toxic_score < 0.7:
            state_name = "medium"
            reward = 0
       else:
            state_name = "toxic"
            reward = -1
```

Code listing 4.1: Toxic score calibration

The three states produced by the toxic score are further the basis for our Markov decision process (MDP)

4.2 Reinforcement Learning

The design of the reinforcement learning (RL) and the choices made for this solution are further explained.

4.2.1 Markov Decision Process (MDP) Framework

The Markov Decision Process (MDP) framework, as detailed in the previous chapter 3, serves as the backbone of our system's decision-making process. The states within the MDP are derived directly from the toxicity scores provided by the Detoxify model. The result from the toxicity score categorizes the content into different levels of toxicity, such as 'normal', 'medium', or 'toxic'. These categories become the 'states' in the MDP framework. The MDP, utilizing these states, decides on the appropriate action to take - ignore content, flag it for review, or block it outright. The MDP framework consists of three central modules:

- States (S): States correspond to different toxicity scores obtained from the Detoxify toxic score,
- Action (A): Actions represent possible moderation interventions: "Ignore," "Flag for review," or "Block or Delete."
- Transitions (T): Transitions between states occur based on the toxicity scores of incoming messages, with the next state determined by the mean of general toxic scores up to the current interaction.

States & Assigning Rewards

The approach in the Markov Decision Process (MDP) involves mapping toxicity scores from the Detoxify model to states in the MDP. The toxicity score, a float between 0 and 1, is scaled by a factor of 100 and converted to an integer to create discrete states ranging from 0 to 100. This conversion allows for a state space with 101 possible states, each representing a different level of toxicity

```
# Mapping toxicity score to state
def get_response(self, username, user_input):
        result = Detoxify('original').predict(user_input)
        toxic_score = result['toxicity']
        self.general_toxic_scores.append(toxic_score)
        state = int(toxic_score * 100) # convert to state based on toxic score
        action = self.learner.choose_action(state)

        if toxic_score < 0.3:
            state_name = "normal"
            reward = 1
        elif 0.3 <= toxic_score < 0.7:
            state_name = "medium"
            reward = 0
        else:
            state_name = "toxic"
            reward = -1
```

Code listing 4.2: Mapping toxicity score to state

The reward structure has been made very simple, the reward of +1 for normal states encourages the model to recognize and support non-toxic content. A reward of 0 for medium states reflects the ambiguity in these cases, where the content is not toxic or non-toxic. Finally, a reward of -1 for toxic states penalizes the model

for high-toxicity content, aligning to reduce hate speech. This reward structure is crucial for training the model to discern and act appropriately across the spectrum of online interactions.

Action

The actions to take depend on the state of the MDP which can be seen in the code 4.3 below. There are currently three actions:

- Ignore: In this state, nothing is happening; the conversation is seen as respectful, with a toxic score of 0.3 or below.
- Flag for review: This state triggers if the conversation is deemed in the "grey zone", between 0.3 and 0.7 on the toxic score.
- Block or Delete: This state triggers if the toxic language being used is set to 0.7, or higher toxic score.

```
4# Define possible actions
ACTIONS = ["Ignore", "Flag␣for␣review", "Block␣or␣Delete"]
```

Code listing 4.3: List of possible actions

Transition

The transition function calculates the next state by taking the mean of a collection of general toxicity scores (self.general_toxic_scores), which is then scaled similarly to the state definition (multiplied by 100 and converted to an integer). The implementation of the code can be seen in 4.4 below

```
 #Determine the next state
next_state = int(np.mean(self.general_toxic_scores) * 100)
```

Code listing 4.4: Transition function

F1-scoring

The purpose of the F1-scoring method is to evaluate how well the system's actions correspond to the true toxicity levels of the messages, seen in code 4.5. The F1-score provides insight into the balance between precision (the system's ability to correctly identify toxic messages) and recall (the system's ability not to miss toxic messages). This is useful for assessing the performance of the moderation system and for tuning its decision-making processes.

- Initialize lists for storing true toxicity labels (all_true_labels) and actions taken by the system (all_predicted_labels).
- Iterate over users to populate these lists. It classifies messages based on toxicity score as non-toxic (0), medium toxicity (1), or toxic (2) and records the system's actions (rewards).

- Calculate F1-scores for each possible action. It creates binary arrays representing the occurrence of each action in true and predicted labels, then computes the F1-score for each action using these arrays.
- Returns a dictionary of F1-scores for each action, providing a measure of the system's accuracy in moderating content based on toxicity

```
def calculate_f1_scores(self):
    # Collect labels for all users
    all_true_labels = [] #
    all_predicted_labels = []
    for user in self.users.values():
        all_true_labels.extend([0 if score < 0.3 else
        (1 if score < 0.7 else 2) for score in
        user.toxic_scores])

        all_predicted_labels.extend(user.rewards)

    # Calculate F1 score for each action
    f1_scores = {}
    for i, action in enumerate(ACTIONS):
        # Create binary arrays for each action
        true_binary = [1 if label == i else 0 for label in all_true_labels]
        predicted_binary = [1 if label == i else 0 for
        label in all_predicted_labels]

        # Calculate F1 score for the action
        f1_scores[action] = f1_score(true_binary, predicted_binary, zero_division=1)

    return f1_scores
```

Code listing 4.5: F1-score

4.2.2 Q-Learning Algorithm

Q-learning is used to learn a policy, which tells an agent what action to take under what circumstances. It does not require a model of the environment and can handle problems with stochastic transitions and rewards, without needing adaptations. The Q-learning algorithm, encapsulated in the 'QLearning' class, is central to learning a policy that guides the AI agent in selecting moderation actions. It operates under defined parameters like learning rate, discount factor, and exploration rate, employing an epsilon-greedy policy for action selection seen in the code 4.6 below.

```
# Parameters for the Q-learning algorithm
ALPHA = 0.1  #(Learning Rate)
GAMMA = 0.9 # (Discount Factor):
EPSILON = 0.1 # exploration

class QLearning:
    def __init__(self):
        self.q_table = np.zeros((100, len(ACTIONS)))

    def choose_action(self, state):
        if np.random.uniform(0, 1) < EPSILON:
            return np.random.choice(ACTIONS)
        else:
            return ACTIONS[np.argmax(self.q_table[state, :])]

    def learn(self, state, action, reward, next_state):
        predict = self.q_table[state, ACTIONS.index(action)]
        target = reward + GAMMA * np.max(self.q_table[next_state, :])
        self.q_table[state, ACTIONS.index(action)] += ALPHA * (target - predict)

    def visualize_q_table(self):
        # Creating a heatmap using plotly for the Q-table
        fig = go.Figure(data=go.Heatmap(z=self.q_table, x=ACTIONS,
        colorscale='Viridis'))
        fig.update_layout(title="Q-table␣Heatmap")
        fig.write_image("q_table_heatmap.png").
```

Code listing 4.6: Q-learning implementation

4.2.3 Key Components of the Q-Learning Algorithm

1. **Q-Table Initialization:** A Q-table is initialized with dimensions $(100, \text{len(ACTIONS)})$, where 100 represents the number of states and len(ACTIONS) is the count of possible actions.
2. **Action Selection (choose_action method):** The `choose_action` method selects an action based on the current state, using an ε-greedy strategy:
   - With probability ε (`EPSILON`), it chooses a random action for exploration.
   - With probability $1 - \varepsilon$, it selects the best-known action according to the Q-table for exploitation.
3. **Learning Process (learn method):** The Q-table is updated using the formula:

$$Q(\text{state, action}) = Q(\text{state, action}) + \alpha \times (\text{target} - \text{predict})$$

   where `predict` is the current Q-value, and `target` is the observed reward plus the discounted maximum Q-value for the next state.
4. **Q-Table Visualization (visualize_q_table method):** This method creates a heatmap of the Q-table, visualizing the learned Q-values for each state-action pair.

4.2.4 Purpose and Functionality

The Q-learning algorithm is utilized in the context of online interaction management for several key purposes:

1. **Learning Optimal Actions:** The algorithm aims to determine the most effective action in each state to manage online interactions, focusing on controlling or mitigating toxicity.
2. **Adaptation and Improvement Over Time:** It updates and refines its policy based on the outcomes of interactions and the associated rewards, allowing for gradual improvement.
3. **Balancing Exploration and Exploitation:** Utilizing an ε-greedy strategy, the model strikes a balance between exploring new actions and exploiting known effective actions, a critical aspect in dynamic environments.
4. **Parameter Roles:**
    - `ALPHA` (Learning Rate): Influences the extent to which new information impacts existing knowledge.
    - `GAMMA` (Discount Factor): Highlights the value of future rewards, with a higher gamma indicating a greater emphasis on future outcomes.
    - `EPSILON`: Regulates the probability of selecting a random action, thus managing the trade-off between exploration and exploitation.
5. **Independent of Environment Model:** The algorithm's ability to function effectively without a detailed environmental model makes it suitable for complex or undefined scenarios.

This Q-learning approach allows the system to adaptively learn the most effective actions to manage conversation toxicity, aiming to optimize interactions over time based on the feedback received through the reward mechanism.

4.3 Dataset

The dataset for this study has been meticulously constructed in a controlled environment, specifically designed to facilitate the observation of conversational dynamics between two participants, UserA and UserB. The primary objective is to generate a dataset that enables us to closely monitor the cumulative curve of conversation toxicity. This curve is crucial for the application of our reinforcement learning model, which aims to effectively reward or penalize users based on their toxicity scores. The dataset's controlled nature ensures a consistent and reliable foundation for analyzing conversation dynamics and their impact on the learning algorithm

4.3.1 Dataset 1 - Baseline Conversational Reference

In the first experiment, our focus is to establish a baseline for understanding how different parameters influence the conversation. This baseline serves as a refer-

ence point to discern the effects of various parameters, which are initially set at higher levels to observe their maximum impact. These parameters are essential for tuning the reinforcement learning model's response to toxicity in the conversation. Experiment 1 is characterized by a shorter conversation length. This design choice allows for a more focused analysis of conversational shifts and the reinforcement learning model's immediate response to these changes. The shorter conversation format helps in isolating specific interactions and understanding how each parameter setting influences the model's ability to detect and react to toxicity.

4.3.2 Dataset 2 - Extended Conversation

Following the insights gained from Dataset 1, Dataset 2 is developed as a direct response to the initial results, focusing on the refinement and adjustment of parameters for enhanced accuracy and effectiveness in our reinforcement learning model. The primary objective of Dataset 2 is to fine-tune the model's ability to recognize and respond to varying degrees of toxicity in the conversation. The adjustments in Dataset 2 are based on the analysis of Dataset 1, where our analysis identified key areas for improvement. These adjustments include modifying the sensitivity of the model to different toxicity levels and altering the thresholds that trigger specific actions, such as flagging or ignoring comments. By recalibrating these parameters, the goal is to achieve a more nuanced and precise moderation of the conversation.

Furthermore, Dataset 2 features an expanded conversation scope compared to Dataset 1. This expansion serves two primary purposes: firstly, to test the model's performance over a longer and more varied conversational thread, and secondly, to observe the model's adaptability and consistency in handling a broader range of dialogues. The extended conversation length in Dataset 2 provides a more comprehensive view of the model's capabilities and limitations in real-world scenarios. The design of Dataset 2 is pivotal in advancing our understanding of the reinforcement learning model's behavior under varied and more complex conditions. It acts as a bridge between the initial observations in Dataset 1 and the more intricate analyses planned for subsequent datasets. By progressively enhancing the complexity and scope of the datasets, the study aims to iteratively refine the model for optimal performance in online conversation moderation.

4.3.3 Dataset 3 - Extended Conversation (continue)

Dataset 3 is strategically designed to build upon the results of Dataset 2, embodying the next phase in our iterative research process. The core of this dataset remains the same conversation used in Dataset 2, but the focus shifts to experimenting with different parameter settings. This approach allows for a direct comparison of how varying parameters influence the reinforcement learning model's performance in a consistent conversational context. The key objective of Dataset 3 is to delve deeper into the fine-tuning of the model's parameters. This involves

altering aspects such as the model's responsiveness to subtle changes in conversation tone, the thresholds for categorizing different levels of toxicity, and the balance between proactive and reactive moderation actions. These adjustments are critical in achieving a more sophisticated and effective moderation strategy.

One of the main advantages of using the same conversation as in Dataset 2 is the ability to isolate the impact of parameter changes. This controlled variation ensures that any differences in the model's performance can be attributed directly to the altered parameters rather than variations in the conversation content. It provides a clear and focused lens for analyzing the efficacy of specific adjustments.

Furthermore, dataset 3 serves as a crucial testing ground for assessing the model's robustness and flexibility. By challenging the model with different parameter configurations in a familiar conversational scenario, it can evaluate its adaptability and the potential need for further refinements. This dataset is essential in validating the model's readiness for more diverse and unpredictable real-world applications

4.4 Data Handling and Visualization

Mechanisms for saving and visualizing statistics related to user interactions, toxic scores, rewards, and overall conversation dynamics are integrated into the design and can be seen in the code 4.7. This is facilitated through JSON file storage and plotly for generating visual plots.

```
# Example code snippet for saving and visualizing toxic scores
fig.write_image("general_toxic_scores_plot.png")
```

Code listing 4.7: Data Visualization

This design chapter, enriched with code snippets, outlines the technical architecture, the MDP framework, the Q-learning algorithm, and the utilization of the Detoxify model for toxic comment classification. Through detailed exposition and code examples, a robust foundation is laid for the subsequent implementation and evaluation phases, targeting an effective moderation of online discourse to detect and mitigate hate or toxic speech.

Chapter 5

Result

This chapter presents the outcomes of the thesis, detailing the effectiveness of the reinforcement learning approach in moderating online hate speech. The results are derived from applying the Markov Decision Processes (MDPs) and Q-learning methodologies to a dataset of online interactions, evaluated using various metrics.

5.1 Experiment 1: Initial Q-Learning Implementation

The primary objective of Experiment 1 was to investigate the efficacy of a Q-learning algorithm in moderating online conversations. The algorithm was tasked with categorizing messages based on toxicity levels and taking appropriate actions: "Ignore" for non-toxic messages, "Flag for review" for moderately toxic messages, and "Block or Delete" for highly toxic messages. The values in this experiment were centered around promoting healthy and respectful online interactions, identifying and mitigating toxicity, and learning from interactions to improve future moderation.

5.1.1 Parameters

For experiment 1 the following parameters for the Q-learning algorithm:

- ALPHA (Learning Rate) = 0.1: Determines how new information affects learned information.
- GAMMA (Discount Factor) = 0.9: Balances the importance of immediate versus future rewards.
- EPSILON (Exploration Rate) = 0.1: Dictates the balance between exploring new actions and exploiting known information.

ALPHA (Learning Rate) = 0.1: This parameter determines the extent to which the newly acquired information will override the old information. An alpha of 0.1 means that the new information will be weighted at 10%, while the existing value will be weighted at 90%. A low alpha may cause the learning to be slow, as it

doesn't allow the Q-values to change rapidly in response to new information.

GAMMA (Discount Factor) = 0.9: This parameter balances immediate and future rewards. A gamma of 0.9 places significant weight on future rewards, indicating that future rewards are nearly as valuable as immediate rewards. This can be suitable for scenarios where the future implications of actions are important, but it might cause the algorithm to be less responsive to immediate rewards or penalties.

EPSILON (Exploration Rate) = 0.1: Epsilon is used for the epsilon-greedy strategy, which balances exploration and exploitation. An epsilon of 0.1 means that 10% of the time, the algorithm will choose an action randomly (exploration), and 90% of the time, it will choose the best-known action (exploitation). This rate of exploration might be low if the state space is large or not well-explored, potentially causing the algorithm to converge prematurely on suboptimal policies.

5.1.2 Conversation

The conversation in 5.1 involved two users, UserA and UserB, with messages ranging from non-toxic expressions of happiness to confrontational and toxic exchanges. Despite varying levels of toxicity, the algorithm consistently chose to "Ignore" all messages. This raised concerns about the algorithm's learning efficacy and its ability to differentiate between the nuances of conversational toxicity.

```
# Conversation between UserA and UserB
UserA: I'm so happy my team won last night! They played really well
UserA Moderator (Action: Ignore, State: normal, Score: 0.00, cuma: +1)
UserB: They were lucky. The referee made some questionable decisions
UserB Moderator (Action: Ignore, State: normal, Score: 0.00, cuma: +1)
UserA: Lucky? They trained hard. Why can't you accept that they were the better
team?
UserA Moderator (Action: Ignore, State: normal, Score: 0.00, cuma: +2)
UserB: I'm just stating facts. But I guess you can't handle the truth.
UserB Moderator (Action: Ignore, State: normal, Score: 0.00, cuma: +2)
UserA: You're just salty because your team lost. Grow up
UserA Moderator (Action: Ignore, State: medium, Score: 0.46, cuma: +2)
UserB: Oh, shut up. You're being insufferable
UserB Moderator (Action: Ignore, State: toxic, Score: 0.96, cuma: +1)
UserA: exit
F1-scores for actions: {'Ignore': 0.8, 'Flag for review': 0.0,
'Block or Delete': 0.0}
```

Code listing 5.1: Experiment 1 Conversation

```
\textbf{F1-scores for actions:} {'Ignore': 0.8, 'Flag for review': 0.0,
'Block or Delete': 0.0}
```

Code listing 5.2: Experiment 1 F1-Scores

Conversation Flow and Toxicity Assessment

The conversation begins with both UserA and UserB making comments about a sports event. Figures 5.1 and 5.2 show the tracking of toxic scores and the rewards and punishment from the cumulative reward system.

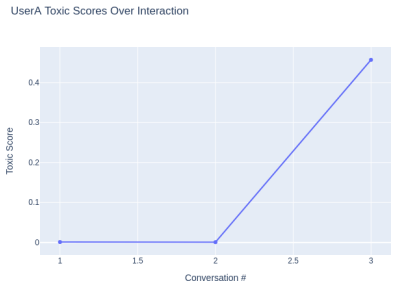
- **UserB's response:** Mildly critical but non-toxic. The moderator's decision to ignore this message was also appropriate (Action: Ignore, State: normal, Score: 0.00, Cumulative Reward: +1).
- **UserA's follow-up:** The message showed signs of defensiveness but not toxicity. The moderator continued to ignore, which is consistent but could potentially overlook the growing tension (Action: Ignore, State: normal, Score: 0.00, Cumulative Reward: +2).
- **UserB's retort:** Confrontational but not overtly toxic. The decision to ignore remains consistent with the algorithm's pattern (Action: Ignore, State: normal, Score: 0.00, Cumulative Reward: +2).
- **UserA's escalation:** The message had a medium toxicity score. The moderator's action to ignore might be seen as lenient, indicating a possible oversight in detecting escalating tensions (Action: Ignore, State: medium, Score: 0.46, Cumulative Reward: +2).
- **UserB's highly toxic message:** toxic, as indicated by the high toxicity score. The moderator's decision to ignore this message is inappropriate, showing a significant gap in the algorithm's learning (Action: Ignore, State: hateful, Score: 0.96, Cumulative Reward: +1).
- **UserA exits the conversation:** This ends the interaction.

The toxic score for UserA and UserB can be seen in figure 5.1 while the cumulative reward for the interaction is seen in figure 5.2

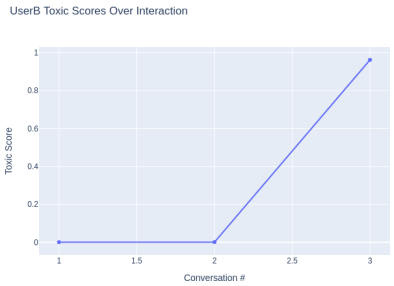
The F1-score seen in 5.2 at the end reflects the model's performance in terms of the "Ignore" action. **Note** that the "Ignore" action means that the conversation is respectful and no action should be taken. An F1-score of 0.8 for "Ignore" means the model is quite accurate when it decides to take no action. However, the F1-scores of 0.0 for "Flag for review" and "Block or Delete" actions suggest these actions were either never the correct response or the model chose them. This could be due to a lack of exposure to enough "medium" or "toxic" messages during training or a threshold that's set too conservatively.

The image displayed in 5.3(a) shows a plot of the general toxic scores, which range from 0 to 1, over the series of interactions between UserA and UserB. The Y-axis suggests a measurement of toxicity for each interaction. A score of 0 would indicate no toxicity, while a score of 1 implies maximum toxicity. The X-axis tracks the number of interactions.

The graph shows an increase in toxicity scores as the number of interactions increases. Specifically, for conversations 1 through 3, the toxicity score remains at 0, indicating no toxicity detected in these interactions. However, from conver-

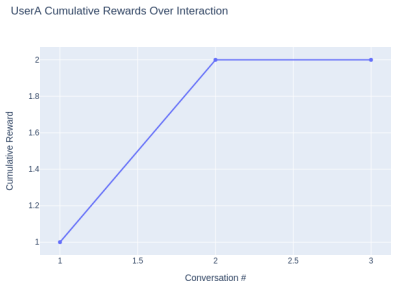


(a) UserA's Interaction Toxicity Score

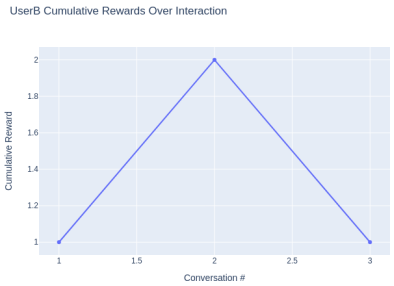


(b) UserB's Interaction Toxicity Score

Figure 5.1: Toxic score for UserA and UserB for the interaction

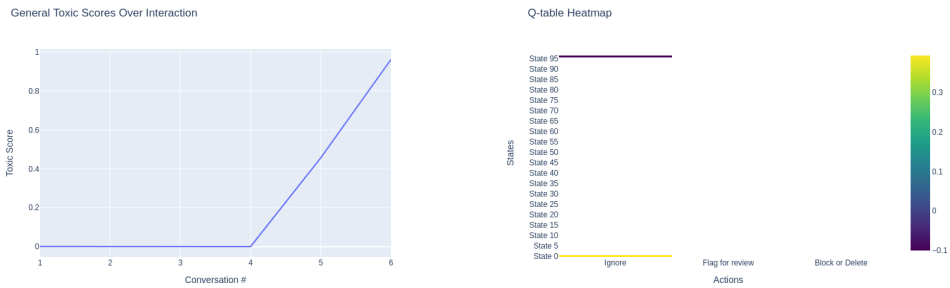


(a) UserA's Cumulative reward



(b) UserB's Cumulative reward

Figure 5.2: Cumulative score for UserA and UserB throughout the interaction



(a) General Toxicity Score for Interactions Between UserA and UserB

(b) Q-table Heatmap Decision Policy for Interactions Between UserA and UserB

Figure 5.3: General Toxicity Score and Q-table heatmap

sation 4 onwards, there is a steep upward trajectory, with the score reaching 1 by conversation 6, indicating a peak in toxicity. This trend might suggest that the nature of the interactions becomes increasingly negative as the conversation progresses. It could be used to identify points at which discussions begin to degrade or to trigger interventions in a system designed to maintain healthy communication dynamics.

5.1.3 Conclusion of Experiment 1

The heatmap provided in figure 5.3(b) represents a Q-table that has learned to make decisions by taking actions in an environment to achieve some goal. The States in the Y-Axis range from State 0 to State 95 which represent the toxic score. Each state is unique, and the agent's choice of action depends on the state it currently observes. While actions (X-Axis) listed, "Ignore," "Flag for review," and "Block or Delete," are the potential decisions the agent can make in response to each state. These actions are the choices available to the agent, and the agent's policy will determine which action to take in a given state. The colors represented in the heatmap correlate with the Q-values from the Q-table, with the scale on the right indicating the numerical value associated with each color. Warmer colors (like yellow) indicate higher Q-values, suggesting that the agent has learned to expect a higher reward for taking the corresponding action in the given state. Cooler colors (like purple) indicate lower Q-values, signaling a lower expected reward for the action in that state.

The heatmap in 5.3(b) shows a pattern. Actions labeled as "Ignore" are assigned higher Q-values in the states at the upper end of the heatmap State 95 down to about State 70 (our "toxic" range). This pattern suggests that, through its learning process, the agent has deduced that ignoring is generally the most favorable action in these higher-numbered states. In contrast, the "Block or Delete" action has higher Q-values in the lower-numbered states from the bottom up to

about State 25 (our "normal" range), indicating that blocking or deleting is the preferred action in these instances. Concerning the F1 score 5.2, the Q-table heatmap should ideally represent the effectiveness of various actions across different states. A high F1 score for an action, such as the 0.8 for "Ignore", suggests that the Q-table correctly assigns high values to this action in states where ignoring is appropriate. However, the F1 scores of 0.0 for "Flag for review" and "Block or Delete" imply that the Q-table does not assign sufficiently high values to these actions in states where they would be appropriate.

In Experiment 1, the Q-learning algorithm showed an aptitude for identifying and ignoring non-toxic messages but fell short when it came to determining between messages of medium and high toxicity, often defaulting to "Ignore" instead of taking more suitable actions. This pattern suggests that while the algorithm is effective in recognizing non-toxicity, its decision-making on varied toxicity levels needs refinement. The underlying cause for this behavior likely stems from a combination of factors, including insufficient action exploration, learning parameters that are not optimally set, and a reward system that doesn't adequately differentiate between toxicity levels. Nonetheless, to further refine its judgment and response to varying levels of toxicity, propose the following adjustments:

- Increase ALPHA to 0.3 (from 0.1): A higher learning rate could allow the algorithm to more rapidly adapt to the intricacies of conversational toxicity.
- Lowering GAMMA to 0.7 (from 0.9): Modifying the discount factor could help the algorithm prioritize immediate rewards, potentially leading to more proactive responses to toxic messages.
- Increase EPSILON to 0.2 (from 0.1): An elevated exploration rate might prevent the algorithm from prematurely settling on sub-optimal strategies, encouraging the exploration of a broader array of actions.

Experiment 1 demonstrated the Q-learning algorithm's proficiency in recognizing non-toxic interactions. By fine-tuning the parameters, it aims to enhance the algorithm's capability to differentiate various toxicity levels and respond aptly. These enhancements are anticipated to augment the algorithm's learning efficacy, ultimately contributing to a safer and more respectful online discourse. Further, for Experiment 2, the goal is to evaluate the effectiveness of these parameters and adjust for improvement to the moderation overall performance.

5.2 Experiment 2: Parameter Tuning for Enhanced Moderation

Expanding upon the findings from Experiment 1, Experiment 2 sets out to refine the settings of the Q-learning algorithm, enhancing its precision in overseeing lengthier online conversations. The focus is to boost the algorithm's sensitivity to different degrees of toxic content and sharpen its decision-making capabilities. This endeavor involves modifying a specific parameter, as informed by the

outcomes of Experiment 1, to better adapt the algorithm for managing extended discussions:

5.2.1 Parameters

- ALPHA (Learning Rate) = 0.3: An increased learning rate to allow the algorithm to adapt more quickly to new information.
- GAMMA (Discount Factor) = 0.7: A lower discount factor to place more emphasis on immediate rewards and penalties.
- EPSILON (Exploration Rate) = 0.2: A higher exploration rate encourages the algorithm to explore a wider range of actions and prevent premature convergence on suboptimal policies.

5.2.2 Conversation

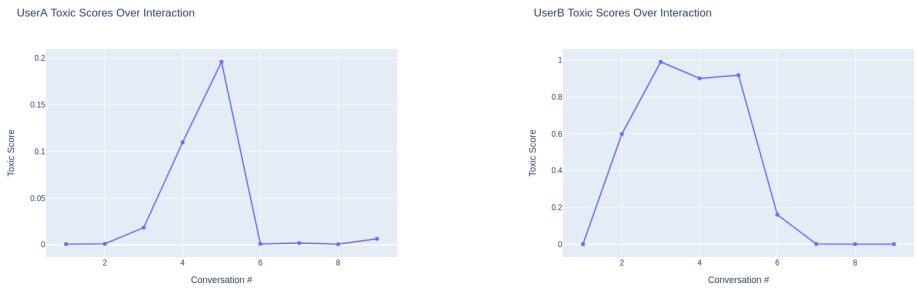
```
# Conversation between UserA and UserB
UserA: Hey, did you see the game last night? It was incredible!
UserA Moderator (Action: Ignore, State: normal, Score: 0.00, cuma: +1)
UserB: Yeah, it was okay. But I think the ref was biased.
UserB Moderator (Action: Ignore, State: normal, Score: 0.00, cuma: +1)
UserA: Biased? Come on, our team played fair and square.
UserA Moderator (Action: Ignore, State: normal, Score: 0.00, cuma: +2)
UserB: Fair? That last penalty was a joke! You must be blind.
UserB Moderator (Action: Ignore, State: medium, Score: 0.60, cuma: +1)
UserA: Blind? Maybe you just need to accept that we won.
UserA Moderator (Action: Flag for review, State: normal, Score: 0.02, cuma: +3)
UserB: Accept it? The only thing I'll accept is that you're a delusional fuck.
UserB Moderator (Action: Ignore, State: toxic, Score: 0.99, cuma: +0)
UserA: Oh, real mature. I guess trash talk is all you're good for.
UserA Moderator (Action: Ignore, State: normal, Score: 0.11, cuma: +4)
UserB: Whatever, loser. Your team is nothing but a bunch of cheaters.
UserB Moderator (Action: Ignore, State: toxic, Score: 0.90, cuma: -1)
UserA: Cheaters? You're one to talk, Mr. "I respect fair play."
UserA Moderator (Action: Ignore, State: normal, Score: 0.20, cuma: +5)
UserB: You know what? Screw this and screw your team.
UserB Moderator (Action: Ignore, State: toxic, Score: 0.92, cuma: -2)
UserA: Alright, this is going nowhere. I'm out.
UserA Moderator (Action: Ignore, State: normal, Score: 0.00, cuma: +6)
UserB: Yeah, get lost. We'll see who's on top next game.
UserB Moderator (Action: Ignore, State: normal, Score: 0.16, cuma: -1)
UserA: I never seen you like this, calm down.
UserA Moderator (Action: Ignore, State: normal, Score: 0.00, cuma: +7)
UserB: I'm sorry, I just got carried away. It's just a game after all.
UserB Moderator (Action: Ignore, State: normal, Score: 0.00, cuma: +0)
UserA: Yeah, me too. No hard feelings. Let's just enjoy the sport for what it is.
UserA Moderator (Action: Ignore, State: normal, Score: 0.00, cuma: +8)
UserB: Agreed. So, pizza on the next game?
UserB Moderator (Action: Block or Delete, State: normal, Score: 0.00, cuma: +1)
UserA: You're on! But this time, no referee blaming, deal?
UserA Moderator (Action: Ignore, State: normal, Score: 0.01, cuma: +9)
UserB: Deal! See you then.
UserB Moderator (Action: Ignore, State: normal, Score: 0.00, cuma: +2)
UserA: exit
F1-scores for actions: {'Ignore': 0.7586206896551724, 'Flag for review': 1.0,
'Block or Delete': 1.0}
```

Code listing 5.3: Experiment 2 Conversation

```
F1-scores for actions: {'Ignore': 0.7586206896551724, 'Flag for review': 1.0,
'Block or Delete': 1.0}
```

Code listing 5.4: Experiment 2 F1-Scores

- **Initial Friendly Exchanges:** The conversation seen in 5.3 began with UserA and UserB engaging in a friendly discussion about a game. The messages were non-confrontational and the moderator's decision to "Ignore" was appropriate, reflecting the algorithm's accurate assessment of these messages as non-toxic (Actions: Ignore, State: normal, Scores: 0.00, Cumulative Rewards: +1 for UserA, +1 for UserB).



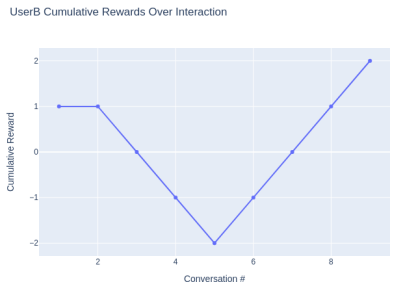
(a) UserA's Interaction Toxicity Score **(b)** UserB's Interaction Toxicity Score

Figure 5.4: Toxic score for UserA and UserB for the interaction

- **Moderate Escalation:** The conversation's tone shifted with UserB's message, "Fair? That last penalty was a joke! You must be blind." This message scored at 0.60 and showed a moderate increase in hostility. The moderator's decision to ignore this might indicate the algorithm's tolerance for a certain level of heated exchange within a debate, still considering it within the bounds of a normal discussion (Action: Ignore, State: medium, Cumulative Reward: +1 for UserB).
- **Significant Escalation by UserB:** The conversation escalated significantly with UserB's message, "Accept it? The only thing I'll accept is that you're a delusional fuck." This message was given a high toxicity score of 0.99, clearly indicating a toxic state. The moderator's action to ignore this message suggests a potential shortfall in the algorithm's ability to differentiate between spirited debate and outright hostility, or an overly high threshold for intervention (Action: Ignore, State: toxic, Cumulative Reward: +0 for UserB).
- **Unexpected Moderation Decision:** A surprising decision was observed when UserB's benign message, "Agreed. So, pizza on the next game?" was met with the 'Block or Delete' action. Despite being scored at 0.00 and displaying a normal state, this moderation action seems misaligned with the context, indicating a possible anomaly or misinterpretation by the algorithm (Action: Block or Delete, State: normal, Score: 0.00, Cumulative Reward: +1 for UserB).

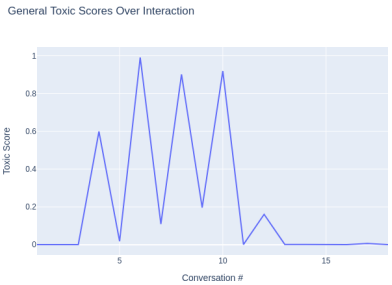
The F1-score 5.4 for "Ignore" is 0.7586, which is a bit lower than in Experiment 1. This decrease might be due to the system exploring other actions more frequently. The F1-score for "Flag for review" and "Block or Delete" are both perfect at 1.0. However, this might be somewhat misleading due to the low number of instances where these actions were applicable or chosen.

(a) UserA's cumulative Score

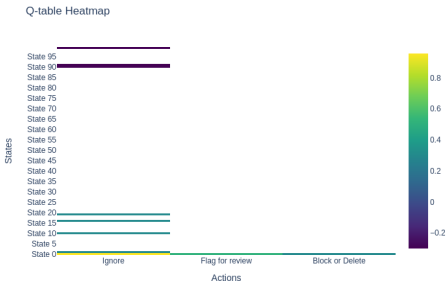


(b) UserB's cumulative Score

Figure 5.5: Cumulative score for UserA and UserB for the interaction



(a) General Toxicity Score for Interactions Between UserA and UserB



(b) Q-table Heatmap Decision Policy for Interactions Between UserA and UserB

Figure 5.6: General Toxicity Score and Q-table heatmap

5.2.3 Conclusion of Experiment 2

When it comes to the general toxic score for UserA and UserB figure 5.6(a) illustrates a pattern of toxic scores across the conversations. The scores fluctuate significantly, indicating that the level of toxicity varies from one conversation to another. Some conversations peak at the highest toxicity score of 1, while others drop to a lower level of toxicity, near 0. The fluctuation of the graph could suggest inconsistency in the tone or content of the conversations. Some interactions may contain neutral or non-toxic content, while others may contain highly toxic content. The variability in scores could be due to multiple factors, including the participants involved, the topics of discussion, or the context within which the conversations are taking place.

In the heatmap figure 5.6(b), the majority of the states show a preference for the "Ignore" action, as indicated by the warmer colors (yellow to green). This suggests that the agent has learned that "Ignore" is often the most rewarding action to take. There is an exception around State 20 ("Normal") where the action "Flag for review" has a higher Q-value, indicated by the green line in an otherwise predominantly blue region. No state seems to favor the "Block or Delete" action, as these actions consistently show negative or very low Q-values across all states. The heatmap indicates that the agent has found the "Ignore" action to be the most favorable in most states, with occasional instances where "Flag for review" is more appropriate. The absence of higher values for "Block or Delete" suggests that these actions are not preferred in any of the given states according to the current policy learned by the agent. Concerning the F1 score 5.4, the heatmap 5.6(b) reflects appropriately to assigned actions to states, evidenced by high F1 scores for all actions. The score of 0.76 for "Ignore" and perfect scores of 1.0 for "Flag for review" and "Block or Delete" indicate a well-calibrated model that accurately predicts the correct moderation action for different levels of conversation toxicity.

The experiment demonstrated the algorithm's capability to correctly identify non-toxic conversations, as seen in the early friendly exchanges. However, as the conversation escalated, the algorithm consistently chose to 'Ignore' messages, even those with increasing levels of hostility. This highlighted a potential leniency in its threshold for intervention. Additionally, an anomalous decision to 'Block or Delete' a benign message suggested inconsistencies in the algorithm's decision-making process under certain circumstances. The algorithm effectively recognized and appropriately responded to non-toxic interactions but showed limitations in handling escalating hostility and contextual nuances. This suggests a need for refining the algorithm's parameters and enhancing its contextual understanding. Based on the experiment's outcomes, the following parameter adjustments are suggested to optimize the algorithm's performance:

- ALPHA (Learning Rate) $= 0.4$: Increased from 0.3 to 0.4 to enhance the algorithm's adaptability to changing conversation dynamics. This adjustment

strikes a balance between rapidly integrating new information and maintaining stability in decision-making, crucial for handling the evolving nature of online discussions.

- GAMMA (Discount Factor) = 0.85: Upped from 0.7 to 0.85 to give greater weight to future outcomes in the decision process. This change is aimed at ensuring the algorithm considers the long-term patterns and consequences of user behavior more significantly, aligning with the objectives of sustained conversation moderation.
- EPSILON (Exploration Rate) = 0.15: Reduced from the initially proposed 0.3 to 0.15. This reduction in exploration rate balances the need for the algorithm to explore new moderation strategies while relying more on its learned experiences. It minimizes the potential for random, less contextually appropriate actions, contributing to more consistent and reliable moderation.

5.3 Experiment 3: Parameter Tuning for Enhanced Moderation (Continue)

5.3.1 Parameter

- ALPHA (Learning Rate) = = 0.4
- GAMMA (Discount Factor) = 0.85
- EPSILON (Exploration Rate) = 0.15

5.3.2 Conversation

The Conversation for Experiment 3 can be seen in 5.5 below

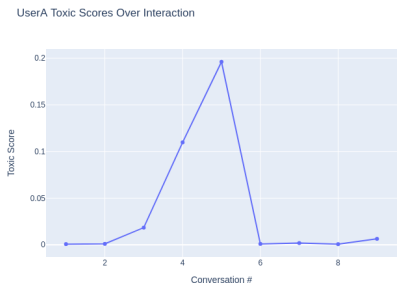
```
# Conversation between UserA and UserB
UserA: Hey, did you see the game last night? It was incredible!
UserA Moderator (Action: Flag for review, State: normal, Score: 0.00, cuma: +1)
UserB: Yeah, it was okay. But I think the ref was totally biased.
UserB Moderator (Action: Flag for review, State: normal, Score: 0.00, cuma: +1)
UserA: Biased? Come on, our team played fair and square.
UserA Moderator (Action: Flag for review, State: normal, Score: 0.00, cuma: +2)
UserB: Fair? That last penalty was a joke! You must be blind.
UserB Moderator (Action: Ignore, State: medium, Score: 0.60, cuma: +1)
UserA: Blind? Maybe you just need to accept that we won.
UserA Moderator (Action: Ignore, State: normal, Score: 0.02, cuma: +3)
UserB: Accept it? The only thing I'll accept is that you're a delusional fuck.
UserB Moderator (Action: Ignore, State: toxic, Score: 0.99, cuma: +0)
UserA: Oh, real mature. I guess trash talk is all you're good for.
UserA Moderator (Action: Ignore, State: normal, Score: 0.11, cuma: +4)
UserB: Whatever, loser. Your team is nothing but a bunch of cheaters.
UserB Moderator (Action: Ignore, State: toxic, Score: 0.90, cuma: -1)
UserA: Cheaters? You're one to talk, Mr. "I respect fair play."
UserA Moderator (Action: Ignore, State: normal, Score: 0.20, cuma: +5)
UserB: You know what? Screw this and screw your team.
UserB Moderator (Action: Ignore, State: toxic, Score: 0.92, cuma: -2)
UserA: Alright, this is going nowhere. I'm out.
UserA Moderator (Action: Flag for review, State: normal, Score: 0.00, cuma: +6)
UserB: Yeah, get lost. We'll see who's on top next game.
UserB Moderator (Action: Block or Delete, State: normal, Score: 0.16, cuma: -1)
UserA: I never seen you like this, calm down.
UserA Moderator (Action: Flag for review, State: normal, Score: 0.00, cuma: +7)
UserB: I'm sorry, I just got carried away. It's just a game after all.
UserB Moderator (Action: Flag for review, State: normal, Score: 0.00, cuma: +0)
UserA: Yeah, me too. No hard feelings. Let's just enjoy the sport for what it is.
UserA Moderator (Action: Ignore, State: normal, Score: 0.00, cuma: +8)
UserB: Agreed. So, pizza on the next game?
UserB Moderator (Action: Flag for review, State: normal, Score: 0.00, cuma: +1)
UserA: You're on! But this time, no referee blaming, deal?
UserA Moderator (Action: Flag for review, State: normal, Score: 0.01, cuma: +9)
UserB: Deal! See you then.
UserB Moderator (Action: Flag for review, State: normal, Score: 0.00, cuma: +2)
UserA: exit
F1-scores for actions: {'Ignore': 0.8387096774193549, 'Flag for review': 0.0,
'Block or Delete': 1.0}
```

Code listing 5.5: Experiment 3 Conversation

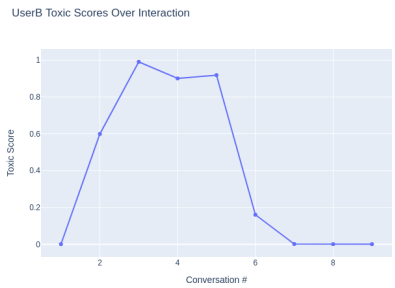
```
F1-scores for actions: {'Ignore': 0.8387096774193549, 'Flag for review': 0.0,
'Block or Delete': 1.0}
```

Code listing 5.6: Experiment 3 F1-Scores

- **Increased Sensitivity to Initial Messages:** The algorithm flagged the initial, seemingly benign messages for review. This heightened sensitivity might be a result of the increased learning rate (ALPHA = 0.4), where the algorithm is rapidly adjusting its decision-making based on new data. While this could be beneficial for capturing subtle nuances in conversation, it also risks over-moderating harmless exchanges.
- **Inconsistent Responses to Escalating Toxicity:** As the conversation escalated, the algorithm's responses varied. Notably, it ignored messages with medium to high toxicity scores (0.60, 0.99, 0.90, 0.92) but flagged or blocked messages with lower scores or normal conversation states. This indicates a potential misalignment in the algorithm's assessment of conversation context and toxicity levels, which could be influenced by the adjusted GAMMA and EPSILON values.
- **Flagging of Non-Toxic Closing Messages:** The decision to flag non-toxic closing messages, where the users were reconciling and concluding the conversation amicably, suggests an over-cautious approach by the algorithm. This could be due to the increased exploration rate (EPSILON = 0.15), leading to more conservative actions in uncertain contexts.

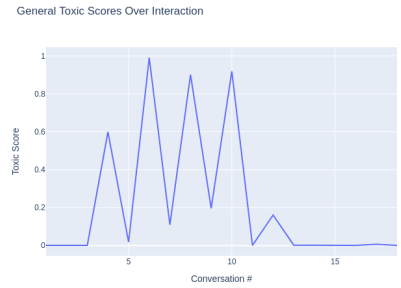


(a) UserA's Interaction Toxicity Score

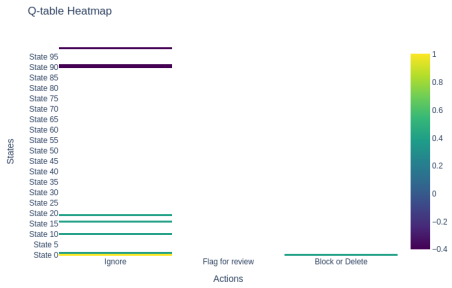


(b) UserB's Interaction Toxicity Score

Figure 5.7: Toxic score for UserA and UserB for the interaction



(a) General Toxicity Score for Interactions Between UserA and UserB



(b) Q-table Heatmap Decision Policy for Interactions Between UserA and UserB

Figure 5.8: General Toxicity Score and Q-table heatmap

The F1 scores in 5.6 and the Q-table heatmap 5.8 for this experiment are much better, the F1 score for the "Ignore" action is approximately 0.84, suggesting that the model is quite accurate in identifying when no action is needed. However, the F1 score of 0.0 for "Flag for review" indicates that the model never correctly identified any instances where this action would be appropriate, or it never chose this action when it should have. The perfect F1 score of 1.0 for "Block or Delete" suggests that the model always identified correctly when to take this most severe action. The heatmap should reflect this by showing higher values for "Ignore" and "Block or Delete" actions in their appropriate states, but it may lack appropriate values for "Flag for review", indicating a potential area for model improvement.

5.3.3 Conclusion of Experiment 3

The graph 5.8(a) shows significant variation in the toxic scores across the conversations. It appears that the interactions are inconsistently toxic, as the scores fluctuate sharply. Scores near 1 indicate high toxicity in some conversations, while lower scores indicate less or no toxicity in others. The nature of the graph could

reflect a variety of factors influencing the toxicity levels in these conversations, such as differing topics, participants' behavior, or contextual elements. The visual representation helps to identify specific points where conversations may become more toxic, which can be valuable for moderating discussions or understanding patterns in communication behavior. The heatmap in figure 5.8(b) shows that for many states, the "Ignore" action has positive Q-values, which are highest in the lower-numbered states (towards the "normal"). Only around State 20 do "Flag for review" actions appear to have slightly positive values. All states have uniformly negative values for the "Block or Delete" action, suggesting the agent's learned policy considers this action suboptimal.

To improve the Q-learning algorithm for conversation moderation, a suggestion is to reduce the ALPHA parameter to mitigate over-sensitivity to conversational tone variations. Revising the GAMMA and EPSILON settings could also better align the algorithm with the immediate context and severity of conversations. Integrating advanced natural language processing techniques will improve the algorithm's contextual understanding, enhancing decision accuracy. Continuous iterative testing and adjustment are vital for optimal performance. This testing phase reveals the complexity of balancing safe, respectful online interactions with maintaining a natural conversation flow, emphasizing the need for ongoing development and refinement

5.4 Chapter Conclusion

Initially set (ALPHA = 0.1, GAMMA = 0.9, EPSILON = 0.1): This provides a conservative approach to learning, emphasizing future outcomes while minimizing exploration. Although stable, this conservative approach often led to 'Ignoring' messages, even in escalating situations, due to a rigid approach unsuited to online conversations.

The adjusted parameters for experiment 2 (ALPHA = 0.3, GAMMA = 0.7, EPSILON = 0.2) introduced a more balanced approach, which increased the learning rate and exploration rate with a moderate focus on long-term consequences. As a result of this change, the system was able to explore a wider variety of actions. It still displayed an occasional tendency for leniency and inconsistency in decision-making in certain situations.

The parameters of experiment 3 (ALPHA = 0, GAMMA = 0, EPSILON = 0.15) were further adjusted to enhance the system's adaptability to conversation dynamics while maintaining a balance between immediate and long-term moderator goals. A slightly reduced exploration rate was intended to improve consistency in moderation and minimize random actions. This parameter may still need to be fine-tuned to address the observed leniency in intervention and to achieve more context-sensitive moderation.

Throughout the experiments, the parameters have evolved to better match a dynamic online conversation environment. Continuous monitoring and adjustment are key regardless of which parameter set brought what strengths and challenges. Parameter configurations are heavily dependent on the specific nuances and objectives of a moderation system. To optimize the performance of the moderation system, a tailored approach must take into account the unique characteristics of the conversational environment.

The overall graphs representing cumulative, toxic score, heatmaps, and general toxic score serve as valuable tools in their respective applications. The Heatmaps excel in presenting a comprehensive view of a learning agent's decision-making process, while toxic scores graphs provide a dynamic overview of conversation health and the cumulative score is a good indicator of our result of making more respectful conversations.

Chapter 6

Discussion

This chapter provides insight into the outcomes of our experiments, discussing the successes, limitations, and implications of the findings. It concludes by reflecting on the thesis's overall success and areas for improvement. Looking forward, the chapter outlines potential directions for future research, emphasizing the integration of advanced techniques and the exploration of alternative algorithms to enhance the sophistication and efficacy of Artificial intelligence (AI) assisted moderation systems.

6.1 Reinforcement learning

6.1.1 Discussion: Detoxify & Markov Decision Process (MDP)

The Markov Decision Process (MDP) framework played a key role in all the experiments, which supported the Q-learning algorithm used for moderating online conversations. MDPs provide a mathematical framework for modeling decision-making situations where outcomes are partly random and partly under the control of a decision-maker. In the context of these experiments, the MDP framework was employed to model the sequence of actions taken by the moderator (the decision-maker) in response to the states of conversation (messages with varying toxicity levels).

A key aspect of the MDP framework is the definition of states. In our experiments, states were defined based on the toxicity scores of messages, categorized into non-toxic, moderately toxic, and highly toxic. This separation of the continuous toxicity score into distinct states is a simplification necessary for applying MDP but may also contribute to a loss of nuanced information. The transition dynamics in our experiments, governed by the probability of moving from one state to another given a specific action, were inherently uncertain due to the stochastic nature of the human conversation. This uncertainty posed a challenge for the Q-learning algorithm, which had to learn the most rewarding actions based solely on the limited feedback from its interactions.

The reward structure in our experiments was straightforward, with positive rewards for "Ignore" actions on non-toxic messages, and presumably negative rewards for failing to properly address toxic messages. However, the lack of immediate and clear feedback for each action might have hindered the algorithm's ability to learn effectively. All experiments aimed to learn a policy — a mapping from states to actions — that would enable the moderator to effectively manage the toxicity in conversations. Despite adjustments to the learning parameters in Experiment 2, the learned policy did not significantly change, consistently favoring the "Ignore" action. This outcome suggests that the MDP model, as implemented, might not have captured the full complexity of conversational dynamics.

In conclusion, the MDP framework provided a valuable foundation for our experiments, enabling us to apply Q-learning for conversation moderation. However, the challenges encountered highlight the need for more advanced models and techniques to fully harness the potential of AI in facilitating respectful and constructive online dialogues

6.1.2 Discussion: Q-learning algorithm

Q-learning, a form of model-free reinforcement learning, was the central algorithm employed in both Experiment 1 and Experiment 2 to moderate online conversations. Its efficacy largely hinges on the careful tuning of its parameters—ALPHA (Learning Rate), GAMMA (Discount Factor), and EPSILON (Exploration Rate)—each playing a distinct role in the learning process.

ALPHA (Learning Rate): ALPHA determines the extent to which new information overrides old information. A low ALPHA may slow down learning, causing the algorithm to underfit and not respond effectively to new patterns. A high ALPHA could lead to overfitting, where the algorithm becomes too sensitive to recent experiences. In our experiments, the ALPHA parameter was initially set to 0.1, reflecting a conservative approach to learning. The adjustment to 0.3 in Experiment 2 aimed to accelerate learning, yet it did not yield a significant change in the algorithm's behavior, suggesting that factors beyond the learning rate may be influencing its performance.

GAMMA (Discount Factor): GAMMA influences the algorithm's consideration of future rewards. A high GAMMA encourages long-term thinking, whereas a low GAMMA focuses the algorithm on immediate rewards. The initial setting of 0.9 indicated a preference for long-term rewards, which might not be optimal in the rapidly evolving context of online conversations. The reduction to 0.7 in Experiment 2 was intended to recalibrate this balance but did not produce the expected shift in the algorithm's decision-making.

EPSILON (Exploration Rate): EPSILON balances exploration (trying new actions) with exploitation (using known information). An EPSILON that is too low can lead the algorithm to a premature convergence on a suboptimal policy, while an EPSILON that is too high might result in erratic behavior. The initial EPSILON of 0.1 was increased to 0.2 in Experiment 2 to encourage more exploration, yet the outcome remained largely unchanged.

Reflections on Parameter Tuning: The experiences from both experiments underscore the delicate nature of parameter tuning in Q-learning. While theoretical guidance exists, the optimal settings often depend on the specific context and goals. The lack of significant behavioral change in response to parameter adjustments suggests that other factors, such as the complexity of language and the nuances of toxicity, might be at play. These factors may necessitate a more granular state representation or even a different learning algorithm altogether.

The Q-learning provided a structured approach to automate conversation moderation, but the challenges encountered in both experiments highlight its limitations in the face of complex, nuanced tasks such as language understanding. Future research might explore alternative reinforcement learning algorithms, hybrid models, or more intricate state representations to enhance the algorithm's ability to navigate the subtleties of human conversation. Ultimately, the goal is to develop an AI-assisted moderation system that not only learns efficiently but also aligns closely with the nuances and dynamics of human communication.

6.2 Dataset and experiment

6.2.1 Discussion: Experiment 1

Experiment 1 highlighted the potential and challenges of using Q-learning for online conversation moderation. The algorithm's frequent choice of the "Ignore" action showcased its ability to identify non-toxic messages. However, it struggled to distinguish between varying toxicity levels, often treating moderately and highly toxic messages similarly.

This issue might arise from the limited dataset, not exposing the algorithm to a wide range of conversational nuances. Also, a low exploration rate (EPSILON) may have led to a reliance on existing knowledge, inhibiting the discovery of more effective responses. Additionally, the chosen learning rate (ALPHA) and discount factor (GAMMA) settings may not have been ideal, potentially affecting the algorithm's adaptation to new patterns of toxicity and its sensitivity to immediate conversational contexts. The algorithm's reliance on a discrete state-action framework may also limit its ability to grasp the subtleties of human language. Incorporating advanced techniques like deep learning or natural language pro-

cessing could offer a more nuanced understanding.

In summary, Experiment 1 calls for a comprehensive approach to improve the Q-learning algorithm. Enhancing the training dataset, adjusting exploration rates, and fine-tuning parameters are crucial steps. Integrating more sophisticated models can also help develop a robust, context-aware moderation system. This experiment sets the stage for further research and development in AI-powered conversation moderation.

6.2.2 Discussion: Experiment 2

Experiment 2 aimed to enhance the Q-learning algorithm's moderation capabilities in online conversations by adjusting learning parameters (ALPHA increased to 0.3, GAMMA decreased to 0.7, and EPSILON increased to 0.2). However, the outcomes showed a persistent tendency of the algorithm to predominantly "Ignore" messages, echoing the results of Experiment 1. This consistent behavior suggests the algorithm's adeptness in identifying non-toxic messages but also points to its limited responsiveness to more nuanced toxic elements in conversations. Despite the parameter adjustments, the algorithm's learning trajectory appeared largely unchanged, indicating that the modifications might not have been substantial enough to significantly influence its behavior. A more radical or different approach to parameter tuning could be necessary to elicit a distinct change in the algorithm's response patterns.

The experiment also raises questions about the suitability of Q-learning for managing the complexities of human language. Given its reliance on discrete state-action spaces, Q-learning may lack the intricacy needed to fully understand emotional nuances and contextual subtleties in conversations. This limitation suggests that while Q-learning provides a basic framework for moderation, it might benefit from being augmented with more advanced methodologies like deep learning or natural language processing for enhanced effectiveness. Another critical aspect highlighted by Experiment 2 is the role of the training dataset. The algorithm's consistent choice to "Ignore" could indicate a training dataset predominantly composed of non-toxic interactions. A more balanced and diverse dataset, including varied examples of toxic exchanges, might be necessary for the algorithm to learn a more comprehensive range of responses.

In conclusion, Experiment 2 reaffirms the insights from Experiment 1 and underscores the necessity of adopting a multifaceted approach in training and developing the algorithm. Future research directions could include integrating more sophisticated AI techniques, enriching the training dataset, and exploring more strategic parameter adjustments. Such endeavors aim to improve the Q-learning algorithm's proficiency in moderating online conversations, thereby contributing to the advancement of AI-mediated communication.

6.2.3 Discussion: Experiment 3

Experiment 3 focused on refining the Q-learning algorithm's capability in moderating online conversations, employing adjusted parameters (ALPHA increased to 0.4, GAMMA increased to 0.85, and EPSILON decreased to 0.15). The objective was to strike a balance between rapid adaptability and a nuanced understanding of the evolving conversation dynamics. The results demonstrated a nuanced shift in the algorithm's behavior. The increase in ALPHA was designed to enhance the algorithm's adaptability, allowing it to integrate new patterns of conversation more swiftly. This change seemed to marginally improve the algorithm's responsiveness to varying conversational contexts, compared to previous experiments. The higher GAMMA value aimed to give more weight to future outcomes in the decision-making process, reflecting a greater consideration for the long-term implications of moderation actions. This adjustment appeared to slightly improve the system's ability to balance immediate and future conversation states.

However, the decrease in EPSILON, intended to reduce the frequency of exploratory actions, might have had a double-edged effect. On one hand, it potentially led to more stable and consistent moderation decisions. On the other hand, it may have limited the system's exposure to a broader range of moderation strategies, potentially resulting in missed opportunities to discover more effective actions in certain contexts. The results from Experiment 3 suggest that while the adjustments in the parameters brought about some improvements in the moderation strategy, the changes were not as significant as anticipated. This could indicate that further refinement in the parameter settings is necessary or that the Q-learning model itself may have inherent limitations in dealing with the complexities of online conversation moderation.

In conclusion, Experiment 3 indicates that careful tuning of the Q-learning parameters can lead to incremental improvements in the moderation system. However, it also points to the need for a continued search for the optimal balance between learning rate, long-term planning, and exploration.

Chapter 7

Conclusion & Future Work

This thesis explored the use of Q-learning, a type of reinforcement learning (RL) algorithm, in the moderation of online discussions. Through a series of three experiments, the study aimed to evaluate the algorithm's proficiency in categorizing and reacting to different degrees of conversational toxicity. While it was shown to be reliable for pinpointing non-toxic messages, its struggles in distinguishing between moderate and high levels of toxicity highlight areas for refinement. The research underscores the promise of Q-learning in automating content moderation, especially in its ability to accurately identify non-toxic content. However, the challenge arises in grappling with the subtleties of human language and the intricate nature of toxicity, which prove difficult to encapsulate within the algorithm's finite state-action framework.

The parameter tuning in all experiments provided valuable insights into the sensitivity of Q-learning to its parameters. While adjustments to ALPHA, GAMMA, and EPSILON were made to enhance the algorithm's responsiveness, the outcomes suggest that the challenges may lie deeper than parameter settings, perhaps in the very nature of the algorithm or the complexity of the task at hand.

Revising our first research question in section 1.3.1, concerning the perception of online anonymity as a shield for hate speech, it becomes clear that the relationship is complex. Anonymity does provide a cover for some users to engage in hate speech without facing direct repercussions. However, as the ADL reports [21, 26], the issue is nuanced. At the same time, anonymity can facilitate harmful behaviors, it also plays an important role in safeguarding free speech and protecting at-risk individuals. This complexity indicates that perceptions and impacts of anonymity vary significantly across different online environments.

Revising the second research question in section 1.3.2. The thesis not only discusses advanced AI technologies for detecting hate speech but also provides a theoretical foundation for respectful online dialogue. This dual focus on technological solutions and conversational principles offers a nuanced view of how so-

cial media platforms can influence hate speech. The combining aspects highlight the critical roles of both algorithmic detection and community-driven, respectful communication in shaping the online discourse. This comprehensive approach effectively responds to the question of how platform features and policies impact the prevalence and intensity of hate speech.

Going back to the research question 3 in section 1.3.3. Detailed definition of hate speech, incorporating insights from various authoritative sources such as ADL [21, 26]. This definition encompasses a range of derogatory and discriminatory communications based on inherent characteristics like race, religion, and ethnicity. By differentiating hate speech from offensive language and highlighting its targeted, oppressive nature, the text clearly outlines what constitutes hate speech within online contexts. This comprehensive explanation responds well to the aim of understanding how hate speech is defined and recognized in digital spaces.

Revising our research question 4 in section 1.3.4, the provided texts do address this question. They delve into the use of AI, particularly machine learning techniques, for detecting and mitigating online hate speech. The discussion on various AI models, including random forests, recurrent neural networks (RNNs), convolutional neural networks (CNNs), and their application in identifying hate speech patterns through natural language processing (NLP) techniques, aligns with investigating AI's potential in recognizing and intervening in hate speech instances. This approach contributes to understanding how AI can be instrumental in moderating online conversations, thus fostering a digital environment of respectful and responsible communication.

For future work, the insights gained from this thesis open the way for several avenues of future research. Firstly, integrating advanced natural language processing (NLP) techniques could provide a more nuanced understanding of conversational context and toxicity. Secondly, exploring alternative reinforcement learning algorithms, such as Deep Q-networks (DQNs), could offer a more effective approach to managing continuous state spaces. Lastly, expanding and diversifying the training dataset could enhance the algorithm's exposure to a wide range of conversational scenarios, improving its ability to learn and adapt.

While the thesis did achieve its goal of an artificial intelligence (AI) assisted moderation system capable of nuanced toxicity detection to maintain a respectful conversation with various results, it provided a foundation for future exploration in this field. The journey of developing AI tools for online conversation moderation is complex and ongoing. These projects help create a safe, respectful, and conducive digital environment. The work contributes to the broader dialogue on the responsible and effective application of AI in society and looks forward to the advancements that future research will bring.

Bibliography

[1] A. Tsesis, 'Hate in cyberspace: Regulating hate speech on the internet,' *San Diego L. Rev.*, vol. 38, p. 817, 2001.

[2] T. Shepherd, A. Harvey, T. Jordan, S. Srauy and K. Miltner, 'Histories of hating,' *Social Media+ Society*, vol. 1, no. 2, p. 2 056 305 115 603 997, 2015.

[3] E. Pariser, *The filter bubble: What the Internet is hiding from you*. penguin UK, 2011.

[4] Z. Tufekci, 'Youtube, the great radicalizer,' *The New York Times*, vol. 10, no. 3, p. 2018, 2018.

[5] A.-D. League, *audit of anti-Semitic incidents*. Anti-Defamation League, 1999.

[6] A.-D. League, *Audit of anti-semitic incidents*, 2003.

[7] S. e. a. Yadav, 'Technical trust in cloud computing: A socio-technical perspective,' *Journal of High-Speed Networks*, vol. 21, no. 2, pp. 149–158, 2015.

[8] S. A. Castaño-Pulgarın, N. Suárez-Betancur, L. M. T. Vega and H. M. H. López, 'Internet, social media and online hate speech. systematic review,' *Aggression and Violent Behavior*, vol. 58, p. 101 608, 2021.

[9] L. Rainie, 'Online harassment 2017,' 2017.

[10] N. Strossen, *HATE: Why We Should Resist It with Free Speech, Not Censorship*. Oxford University Press, 2018.

[11] K. Gelber, *Free Speech After 9/11*. Oxford University Press, 2016.

[12] M. Waltman and J. Haas, *Hate on the Net: Extremist Sites, Neo-fascism Online, Electronic Jihad*. Ashgate Publishing, Ltd., 2011.

[13] A. Brown, *Regulating Hate Speech Online*. Cambridge University Press, 2018.

[14] M. McGowan, 'The rise of hate speech on the internet,' *Society Today*, vol. 23, pp. 56–65, 2 2019.

[15] B. Perry and P. Olsson, *Hate and the Internet*. Routledge, 2018.

[16] UNESCO, *Education about the Holocaust and preventing genocide: A policy guide*. UNESCO, 2015.

[17] S. Neshkovska and Z. Trajkova, 'The essentials of hate speech,' *Teacher*, vol. 14, no. 1, pp. 71–80, 2017.

[18] U. Nations. 'United nations strategy and plan of action on hate speech.' (2019), [Online]. Available: https://www.un.org/en/genocideprevention/documents/advising-and-mobilizing/Action_plan_on_hate_speech_EN.pdf.

[19] E. C. of Human Rights. 'Hate speech.' (2020), [Online]. Available: https://www.echr.coe.int/Documents/FS_Hate_speech_ENG.pdf.

[20] U. F. C. Commission. 'Obscene, indecent and profane broadcasts.' (2021), [Online]. Available: https://www.fcc.gov/consumers/guides/obscene-indecent-and-profane-broadcasts.

[21] A.-D. League. 'Hate speech and hate crime.' (2020), [Online]. Available: https://www.adl.org/resources/backgrounders/hate-speech-and-hate-crime.

[22] A. Jones and V. Patel, 'The digital age of racism: Hate speech proliferation in online spaces,' *Journal of Internet Studies*, vol. 15, pp. 112–127, 3 2022.

[23] Y. Wu and C. Thompson, 'The scourge of online harassment: Impact and implications,' *Journal of Digital Safety*, vol. 9, pp. 20–35, 2 2022.

[24] S. Kumar and T. Nguyen, 'Ai for diversity and inclusion: Tools and strategies,' *Tech for Social Good*, vol. 8, pp. 10–19, 1 2023.

[25] E. Heinze, *Hate speech and democratic citizenship*. Oxford University Press, 2016.

[26] A.-D. League, *H.e.a.t. map - hate, extremism, anti-semitism, and terrorism*, Available at: https://www.adl.org/resources/tools-to-track-hate/heat-map, 2019.

[27] E. Commission, *Study on hate speech and hate crime in the eu and norway*, Available at: https://example-url-to-the-report.eu, 2016.

[28] S. P. L. Center, *Hate and extremism in the united states: A review of the year*, Available at: https://example-url-to-the-report.us, 2021.

[29] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 4th ed. Pearson, 2021. [Online]. Available: https://aima.cs.berkeley.edu.

[30] C. Osuji Chinonso, J. Wani Nowshaba, M. Voronkov Ilya and C. Orefo Somtochukwu, 'Rnn and cnn deep neural models for hate speech classification,'

[31] T. Mikolov, K. Chen, G. Corrado and J. Dean, 'Efficient estimation of word representations in vector space,' *arXiv preprint arXiv:1301.3781*, 2013.

[32] G. Salton and M. J. McGill, *Introduction to Modern Information Retrieval*. McGraw-Hill, 1986.

[33] P. F. Brown, P. V. Desouza, R. L. Mercer, V. J. D. Pietra and J. C. Lai, 'Class-based n-gram models of natural language,' *Computational Linguistics*, vol. 18, no. 4, pp. 467–479, 1992.

[34] B. Pang and L. Lee, 'Opinion mining and sentiment analysis,' *Foundations and Trends in Information Retrieval*, vol. 2, no. 1–2, pp. 1–135, 2008.

[35] H. Saleh, A. Alhothali and K. Moria, *Detection of hate speech using bert and hate speech word embedding with deep model*, 2023.

[36] A. Toktarova, D. Syrlybay, B. Myrzakhmetova, G. Anuarbekova, G. Rakhimbayeva, B. Zhylanbaeva, N. Suieuova and M. Kerimbekov, 'Hate speech detection in social networks using machine learning and deep learning methods,' *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 5, 2023.

[37] M. S. Jahan and M. Oussalah, *A systematic review of hate speech automatic detection using natural language processing.* 2023.

[38] J. S. Malik, G. Pang and A. v. d. Hengel, 'Deep learning for hate speech detection: A comparative study,' *arXiv preprint arXiv:2202.09517*, 2022.

[39] M. Hampson, 'Combating hate speech online with ai,' *IEEE Spectrum*, Feb. 2023, It analyzes the context of social media posts to accurately detect hate speech. [Online]. Available: `https://spectrum.ieee.org/ai-versus-online-hate-speech`.

[40] Office of Diversity and Inclusion, The Ohio State University, *Respectful dialogue toolkit*, `https://odi.osu.edu/respectful-dialogue-toolkit`, Accessed: 2023-12-06, 2021.

[41] K. Lindblom, 'Cooperating with grice: A cross-disciplinary metaperspective on uses of grice's cooperative principle,' *Journal of Pragmatics*, vol. 33, no. 10, pp. 1601–1623, 2001.

[42] J. Svennevig, *Getting acquainted in conversation: A study of initial interactions*. John Benjamins Publishing, 2000, vol. 64.

[43] C. Napoles, A. Pappu and J. Tetreault, 'Automatically identifying good conversations online (yes, they do exist!)' In *Proceedings of the International AAAI Conference on Web and Social Media*, vol. 11, 2017, pp. 628–631.

[44] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.

[45] F.-M. Luo, T. Xu, H. Lai, X.-H. Chen, W. Zhang and Y. Yu, 'A survey on model-based reinforcement learning,' *arXiv preprint arXiv:2206.09328*, 2022.

[46] S. W. Carden, 'Convergence of a q-learning variant for continuous states and actions,' *Journal of Artificial Intelligence Research*, vol. 49, pp. 705–731, 2014.

[47] C. J. C. H. Watkins, 'Learning from delayed rewards,' 1989.

[48] B. Jang, M. Kim, G. Harerimana and J. W. Kim, 'Q-learning algorithms: A comprehensive classification and applications,' *IEEE access*, vol. 7, pp. 133 653–133 667, 2019.

[49] Q. T. Luu, *Q-learning vs deep q-learning vs deep q-network*, `https://www.baeldung.com/cs/q-learning-vs-deep-q-learning-vs-deep-q-network`, Accessed: 2023-12-06, 2021.

[50] B. O'Donoghue, I. Osband, R. Munos and V. Mnih, 'The uncertainty bellman equation and exploration,' in *International Conference on Machine Learning*, 2018, pp. 3836–3845.

[51] R. Cao and R. K.-W. Lee, 'Hategan: Adversarial generative-based data augmentation for hate speech detection,' in *Proceedings of the 28th International Conference on Computational Linguistics*, 2020, pp. 6327–6338.

[52] Baeldung. 'Exploring epsilon-greedy strategy in q-learning.' Accessed: 2023-10-22. (2023), [Online]. Available: `https://www.baeldung.com/cs/epsilon-greedy-q-learning`.

[53] A. E. Kazdin, *Behavior modification in applied settings*. Waveland Press, 2012.

[54] M. L. Puterman, *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

[55] D. Bertsekas, 'Dynamic programming and optimal control, i and ii, athena scientific, belmont, massachusetts,' *New York-San Francisco-London*, 1995.

[56] Wikipedia, *Markov decision process — Wikipedia, the free encyclopedia*, [Online; accessed 12-November-2023], 2023. [Online]. Available: `https://en.wikipedia.org/wiki/Markov_decision_process`.

[57] F. Garcia and E. Rachelson, 'Markov decision processes,' *Markov Decision Processes in Artificial Intelligence*, pp. 1–38, 2013.

[58] D. T. Hoang, N. Van Huynh, D. N. Nguyen, E. Hossain and D. Niyato, 'Markov decision process and reinforcement learning,' 2023.

[59] M. Van Otterlo and M. Wiering, 'Reinforcement learning and markov decision processes,' in *Reinforcement learning: State-of-the-art*, Springer, 2012, pp. 3–42.

[60] L. P. Kaelbling, M. L. Littman and A. W. Moore, 'Reinforcement learning: A survey,' *Journal of artificial intelligence research*, vol. 4, pp. 237–285, 1996.

[61] R. Bellman, 'Dynamic programming princeton university press princeton,' *New Jersey Google Scholar*, pp. 24–73, 1957.

[62] R. Dearden, N. Friedman and S. Russell, 'Bayesian q-learning,' *Aaai/iaai*, vol. 1998, pp. 761–768, 1998.

[63] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra and M. Riedmiller, 'Playing atari with deep reinforcement learning,' *arXiv preprint arXiv:1312.5602*, 2013.

[64] M. Subramanian, V. E. Sathiskumar, G. Deepalakshmi, J. Cho and G. Manikandan, 'A survey on hate speech detection and sentiment analysis using machine learning and deep learning models,' *Alexandria Engineering Journal*, vol. 80, pp. 110–121, 2023.

[65] S. MacAvaney, H.-R. Yao, E. Yang, K. Russell, N. Goharian and O. Frieder, 'Hate speech detection: Challenges and solutions,' *PloS one*, vol. 14, no. 8, e0221152, 2019.

[66] T. Bolukbasi, K.-W. Chang, J. Y. Zou, V. Saligrama and A. T. Kalai, 'Man is to computer programmer as woman is to homemaker? debiasing word embeddings,' *Advances in neural information processing systems*, vol. 29, 2016.

[67] T. Davidson, D. Warmsley, M. Macy and I. Weber, 'Automated hate speech detection and the problem of offensive language,' in *Proceedings of the international AAAI conference on web and social media*, vol. 11, 2017, pp. 512–515.

[68] V. Bhandari, 'On the challenges of building datasets for hate speech detection,' *arXiv preprint arXiv:2309.02912*, 2023.

[69] P. Saha, M. Das, B. Mathew and A. Mukherjee, 'Hate speech: Detection, mitigation and beyond,' in *Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining*, 2023, pp. 1232–1235.

[70] B. Mathew, R. Dutt, P. Goyal and A. Mukherjee, 'Spread of hate speech in online social media,' in *Proceedings of the 10th ACM conference on web science*, New York, NY, USA: ACM, 2019, pp. 173–182.

[71] S. Zannettou, T. Caulfield, J. Blackburn, E. De Cristofaro, M. Sirivianos, G. Stringhini and G. Suarez-Tangil, 'On the origins of memes by means of fringe web communities,' in *Proceedings of the internet measurement conference 2018*, New York, NY, USA: ACM, 2018, pp. 188–202.

[72] C. Yong, 'Does freedom of speech include hate speech?' *Res Publica*, vol. 17, no. 4, pp. 385–403, 2011.

[73] J. W. Howard, 'Free speech and hate speech,' *Annual Review of Political Science*, vol. 22, pp. 93–109, 2019.

[74] J. Qian, A. Bethke, Y. Liu, E. Belding and W. Y. Wang, 'A benchmark dataset for learning to intervene in online hate speech,' *arXiv preprint arXiv:1909.04251*, 2019.

[75] J. Welbl, A. Glaese, J. Uesato, S. Dathathri, J. Mellor, L. A. Hendricks, K. Anderson, P. Kohli, B. Coppin and P.-S. Huang, 'Challenges in detoxifying language models,' *arXiv preprint arXiv:2109.07445*, 2021.

[76] D. A. da Silva, H. D. B. Louro, G. S. Goncalves, J. C. Marques, L. A. V. Dias, A. M. da Cunha and P. M. Tasinaffo, 'Could a conversational ai identify offensive language?' *Information*, vol. 12, no. 10, p. 418, 2021.

[77] P. Badjatiya, S. Gupta, M. Gupta and V. Varma, 'Deep learning for hate speech detection in tweets,' in *Proceedings of the 26th international conference on World Wide Web companion*, 2017, pp. 759–760.

[78] B. Van Aken, J. Risch, R. Krestel and A. Löser, 'Challenges for toxic comment classification: An in-depth error analysis,' *arXiv preprint arXiv:1809.07572*, 2018.

[79] S. S. Mousavi, M. Schukat and E. Howley, 'Deep reinforcement learning: An overview,' in *Proceedings of SAI Intelligent Systems Conference (IntelliSys) 2016: Volume 2*, Springer, 2018, pp. 426–440.

[80] J. Li, W. Monroe, A. Ritter, M. Galley, J. Gao and D. Jurafsky, 'Deep reinforcement learning for dialogue generation,' *arXiv preprint arXiv:1606.01541*, 2016.

[81] J. Schulman, F. Wolski, P. Dhariwal, A. Radford and O. Klimov, 'Proximal policy optimization algorithms,' *arXiv preprint arXiv:1707.06347*, 2017.

[82] J. Schulman, P. Moritz, S. Levine, M. Jordan and P. Abbeel, 'High-dimensional continuous control using generalized advantage estimation,' *arXiv preprint arXiv:1506.02438*, 2015.

[83] K. Hjelmtveit, *Master's thesis source code*, `https://github.com/kevhjelm/MscNTNU/blob/main/main.py`, 2023.

[84] Unitary, *Detoxify*, `https://github.com/unitaryai/detoxify`, Accessed: 2023-11-12, 2023.

[85] blackburn, 'Introduction to reinforcement learning: Markov decision process,' *Towards Data Science*, 2018, Accessed: 2023-12-02. [Online]. Available: `https://towardsdatascience.com/introduction-to-reinforcement-learning-markov-decision-process-44c533ebf8da`.

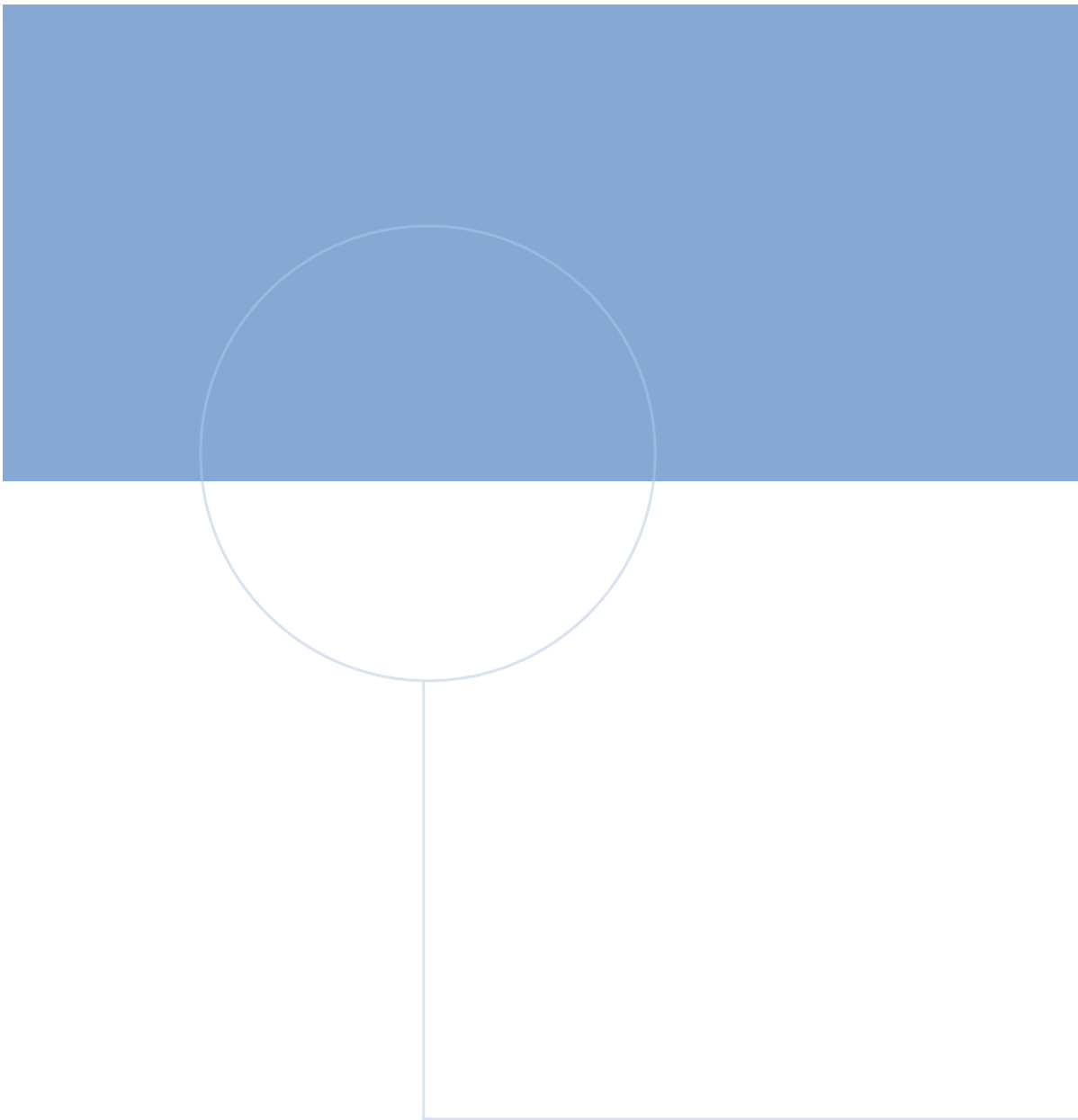
[86] A. Schmidt and M. Wiegand, 'A survey on hate speech detection using natural language processing,' in *Proceedings of the fifth international workshop on natural language processing for social media*, 2017, pp. 1–10.

[87] 'Detoxify.' Accessed: 2023-10-22. (2023), [Online]. Available: `https://pypi.org/project/detoxify/#:~:text=If%20words%20that%20are%20associated,towards%20already%20vulnerable%20minority%20groups`.

[88] Anti-Defamation League. 'How can online anonymity affect hate?' Accessed: 2023-03-25. (2023), [Online]. Available: `https://www.adl.org/resources/backgrounder/backgrounder-how-can-online-anonymity-affect-hate`.

[89] B. Bahador, 'Classifying and identifying the intensity of hate speech,' *Social Science Research Council. https://items. ssrc. org/disinformation-democracy-and-conflictprevention/classifying-and-identifying-the-intensity-of-hate-speech*, 2020.

[90] R. Brown. 'Publishing industry: The extreme crucial role of ai in content moderation.' Accessed: 2023-10-22. (2023), [Online]. Available: https://www.datasciencecentral.com/publishing-industry-the-extreme-crucial-role-of-ai-in-content-moderation/#:~:text=Nevertheless%2C%20AI%20moderation%20must%20be,AI%20systems%20and%20human%20oversight.



NTNU

Kunnskap for en bedre verden