NTNU
Norwegian University of
Science and Technology

Fine tuning BERT for detecting cyber grooming in online chats

Melleby Aarnseth, Simen

Submission date:	June 2023
Main supervisor:	Bours, Patrick, NTNU
Co-supervisor:	Venkatesh, Sushma, AIBA AS

Norwegian University of Science and Technology
Department of Information Security and Communication Technology

Title: Fine tuning BERT for detecting cyber grooming in online chats
Student: Melleby Aarnseth, Simen

Problem description:

Detection of cyber grooming has been an important issue regarding the protection of children on the internet. Many different methods and techniques have been utilized in order to maximize the chances of early recognition. In recent years, machine learning has been seen as a potential solution in order to solve this problem. This thesis will explore how instances of cyber grooming can be detected using the natural language processing model BERT. This will be done by fine tuning already existing models in order to better analyze predatory conversations. This fine tuning is important due to the fact that BERT is trained on meaningful language, not online chats, which tend to be more informal. Additionally, this thesis will further explore how the usage of emojis and internet abbreviations effect BERTs ability to detect cyber grooming. The results achieved from the various models will be compared and examined in order to better understand how BERT can be used to detect cyber grooming as early as possible in chats.

Approved on:
Main supervisor: Bours, Patrick, NTNU
Co-supervisor: Venkatesh, Sushma, AIBA AS

Abstract

This thesis will look into how cyber grooming may be detected through the natural language processing model BERT, with an emphasis on the use of abbreviations and slang present in the chats. To investigate this, several BERT models were trained. These models where trained and tested on different data sets consisting of a varying amount of abbreviations and slang expressions. Through this, BERTs ability to detect cyber grooming based on the prevalence of abbreviations and other informal language forms could be assessed. The findings from this process indicated that BERT was able to detect cyber grooming at a similar rate between data sets where the prevalence of abbreviations and slang was much higher in one compared to the other. This indicated that BERT possesses the ability to understand language quite well despite it being in a more informal form.

Preface

The work for this thesis has been both difficult and rewarding. Along the way I have recieved help and guidance from my supervisors Sushma Venkatesh and Patrick Bours. I would like to thank them for assisting me through out the course of this semester.

Contents

List of Figures

List of Tables

Chapter 1
Introduction

1.0.1 Motivation

Since the invention of the internet, people have been able to connect and relate with others over the entire world. While this could be seen as largely positive it does pose some issues, particularly for the youngest and most vulnerable members of out society. While the internet offers the chance to instantly come into contact with anybody in the world, it also offers an anonymity not available in real life. This can be exploited by people with malicious intents towards children. Since they are still young and do not have the knowledge and experience to deal with such situations appropriately. Studies have shown that at least 20% of children in Norway have received some sort of non-consensual contact or remarks over the internet[**Nova**]. Another poll conducted in Norway on children between the ages of 9 and 17, showed that 7% of these children received unwanted sexual advances and messages on the internet[Eli19].

1.0.2 Scope

This thesis will be exploring how the natural language processing model BERT may be fine tuned in order to better detect instances of cyber grooming in online conversations. Research on this topic has been done before, so the work the main focus of this thesis will be to look at how abbreviations play a part in BERT's ability to detect cyber grooming.

During the start of this thesis, the work focus was too see how both abbreviaitons and emojis played effected BERT when trying to detect cyber grooming. However when reading through the chat data that was used in this study, the prevalence of emojis was quite uncommon in relation to abbrevaitons. Abrreviations appeared far more frequently in the conversations. Due to this issue, the impact of emojis would most likely be insignificant compared to that of abbreviations in BERTs analysis. With this information as a basis, the following research questions where established for this thesis.

1. How may BERT be trained in order to better understand abbreviations and the meaning or context behind them in ongoing chats?
2. How important is the usage of abbreviations for BERT when it is trying to detect cyber grooming? Do any of these more strongly imply that a predatory conversation is or is not taking place?
3. Does replacing an abbreviation with it's original form change how BERT analyzes the chat?

1.0.3 Thesis structure

This thesis is divided into several chapters and a appendix. The second chapter named state of the art reviews relevant background information and previous work that may relate to the topic of this thesis.

The third chapter data gives an overview of the data used to train the BERT models doing this research. It goes into depth about how the data was collected, prepared and sanitized. It also discusses statistics related to the messages and conversations used.

The fourth chapter presents how the initial BERT model was developed and trained. The appendix contains the code related to this process. It also showcases the performance of the model, which wound entail how effective it was in detecting instances of cyber grooming when reading the chats and conversations from the data.

The fifth chapter goes into depth of how the performance of the initial BERT model from the previous chapter could be improved. It also prevents an analysis of how important the usage of abbreviations and other forms of slang are in chats when BERT is trying to analyze data for cyber grooming detection.

The final chapters discuss the results achieved during this research, gives recommandations for future research and a conclusion of the work presented in this thesis.

Chapter 2
State of the art

2.1 Background information

Cyber grooming refers to the manipulative actions undertaken by an individual, typically an adult, through online platforms to establish an emotional connection with a child or young person, with the ultimate intention of exploiting them sexually, psychologically, or for other malicious purposes. It involves a series of deliberate and strategic steps taken by the groomer to gain the trust and confidence of the targeted individual, often by assuming a false identity or using deceptive tactics.

During the grooming process, the groomer employs psychological manipulation techniques to establish emotional rapport and exploit the vulnerabilities of the child. This may involve offering attention, praise, gifts, or even expressing empathy and understanding to create a sense of emotional dependency. Groomers often employ flattery, charm, and deception to lower the child's inhibitions and establish a false sense of trust and friendship. Over time, the groomer may escalate the interaction to involve explicit sexual conversations, sharing inappropriate material, or coercing the child into meeting offline.

2.1.1 Natural Language Processing

Natural Language Processing(NLP) is a branch of artificial intelligence that aims to give computers the ability to understand, interpret and generate human language. It involves the development of algorithms, models, and techniques that facilitate the interaction between computers and human language, enabling tasks such as sentiment analysis, text summarization, and question answering[Edu20].

Natural language processing often involves a range of tasks such as tokenization (splitting text into meaningful units), syntactic parsing (analyzing sentence structure), and semantic analysis (extracting meaning from text). These tasks are often performed using machine learning and deep learning approaches, where models are trained on large annotated datasets to learn the patterns and structures of language.

One of the key challenges in NLP is the inherent ambiguity and complexity of natural language. Words can have multiple meanings, and context plays a crucial role in determining their interpretation. Additionally, language is full of idiomatic expressions, metaphors, and cultural nuances that are easy for humans to understand, but difficult for computers. NLP techniques strive to capture this complexity by incorporating statistical models, linguistic rules, and contextual information to improve the accuracy of language processing systems.

NLP has found numerous applications across various domains. In information retrieval, NLP techniques are used to develop search engines that understand user queries and retrieve relevant documents. In sentiment analysis, NLP enables the analysis of opinions and emotions expressed in text. NLP is also instrumental in machine translation, enabling the automatic translation of text from one language to another, and in chatbots and virtual assistants, where it facilitates natural and interactive conversations between humans and machines.

The advancements in NLP have been driven by the availability of large-scale datasets, powerful computing resources, and breakthroughs in deep learning models, such as recurrent neural networks (RNNs) and transformers. However, challenges still remain. These include ethical considerations and bias in language models.

2.1.2 BERT

BERT is an open source machine learning framework natural language processing. It was designed by Google in 2018. BERT is designed to understand the context and meaning of words by leveraging the power of transformers. Unlike previous models that processed language sequentially from left to right similar to reading, BERT uses a bidirectional approach. This means that is reads and processes sentences from both left to right and right to left. This gives it a deeper understanding of language and sentecne structure compared to natural language processing developed before it[Tou19].

At its core, BERT consists of a deep neural network architecture called a transformer. Transformers utilize self-attention mechanisms, enabling them to weigh the importance of different words in a sentence based on their dependencies. BERT is pre-trained on a large amount of unlabeled text from the internet, allowing it to learn contextualized representations of words. This texts consists of things such as Wikipedia articles and online novels. The pre-training involves two primary tasks: masked language modeling, where certain words are masked and the model predicts them, and next sentence prediction, where the model determines whether two sentences appear in the correct order.

After pre-training, BERT can be fine-tuned on specific downstream tasks, such as sentiment analysis, question answering, or named entity recognition. Fine-tuning involves training BERT on labeled data specific to the task at hand, adjusting its parameters to make more accurate predictions. By fine-tuning BERT on task-specific data, it can adapt to different NLP applications and achieve strong results on a wide range of benchmarks.

The key advantage of BERT lies in its ability to capture the contextual relationships between words, allowing it to generate more accurate and meaningful representations. It overcomes some of the limitations of previous models by considering the surrounding words on both sides of a given word. BERT's success has led to significant advancements in various NLP tasks, improving on the performance of preexisting language models. Its impact has extended beyond academia, with BERT being widely adopted in industry applications and frameworks, and serving as a foundation for further research in NLP.

2.1.3 Transformers

Transformers are a type of deep neural network architecture that have revolutionized the field of natural language processing (NLP). They were introduced in the paper "Attention Is All You Need" in 2017[VSP+17]. Transformers have since become important in many state-of-the-art NLP models, including BERT.

The main innovation of transformers lies in their attention mechanism, which allows the model to weigh the importance of different words or tokens in a sequence based on their dependencies. Unlike previous sequential models, such as recurrent neural networks (RNNs), transformers can process the entire sequence simultaneously, enabling parallelization and capturing long-range dependencies more effectively[VSP+17].

In a transformer, the input sequence is first embedded into a set of high-dimensional vectors, known as embeddings. These embeddings encode the semantic meaning and positional information of the words or tokens. The model then performs multiple layers of self-attention, where each word attends to all other words in the sequence. This attention mechanism assigns weights to the words based on their relevance to the current word, allowing the model to capture contextual relationships.

The self-attention mechanism operates through three key steps: query, key, and value. For each word in the sequence, the query calculates its compatibility with other words (keys) and uses these compatibilities as weights to combine the corresponding values. This process generates a weighted sum, which represents the contextual representation of the word considering its dependencies on other words in the sequence. By performing self-attention across all words in the sequence, transformers can capture complex contextual relationships.

Transformers have several advantages over previous models. They can capture long-range dependencies more effectively, thanks to the self-attention mechanism. Transformers also mitigate the vanishing or exploding gradient problem faced by RNNs, as each word is connected to all other words in the sequence. Furthermore, the parallelizable nature of transformers allows for efficient training on modern hardware.

2.1.4 International Sexual Predator Identification Competition at PAN-2012

The international Sexual Predator Identification Competition was held at PAN-2012. It was a part of a larger annual event which focuses on various aspects of text analysis and forensic language analysis.

The goal of the Sexual Predator Identification Competitions was to develop and evaluate machine learning models to detect sexual predators in online conversations. The participants in the competition where given a large data set consisting of both innocent and predatory chats. The main source of the chat data came from the Perverted Justice Foundation. In these chat logs, adults posed as children in order to lure and expose sexual predators. According to their website, the main motivation behind their actions were the potential arrest of the predators they came in contact with.

With a data set consisting of both innocent and predatory conversations, participants competed amongst themselves by creating the best and most accurate models for detecting cyber grooming. The outcomes of this were positive as advancements were made into methodologies of discovering predators in online conversations. Furthermore this competition helped raise awareness about this important subject, and how machine learning can be used to solve it's related issues.[IC12]

2.2 Related Work

There has been done some research in the fields relating to this thesis, both when for the general fine tuning of BERT and for detecting cyber grooming using natural language processing and machine learning. When it comes to the goal of detecting cyber grooming, several approaches have been taken. These all implementing different techniques and strategies.

Although these works are not directly related to the detection of cyber grooming, natural language processing techniques have been implemented in order to develop methods to detect other instances of unwanted behaviour online such as bullying and harassment.

2.2.1 Structure of a predatory conversation

The linguistic-based empirical analysis approach has been employed to achieve early detection of cyber grooming. In one relevant paper, the authors categorized cyber grooming into six distinct stages and estimated the proportion of each stage within a predatory conversation[GKS12]. These stages include friendship forming, relationship forming, risk assessment, exclusivity, sexual, and conclusion. The analysis revealed that friendship forming constituted the largest portion (approximately 40%) of a sexually predatory chat, while the sexual stage accounted for only 24%. Furthermore, the flow of conversation between stages was found to be non-linear, as chats could jump between stages rather than progress chronologically.

To facilitate their analysis, the researchers utilized a Linguistic Inquiry and Word Count (LIWC) tool available at http://www.liwc.net/. This tool enabled the identification of various word categories that could serve as predictors to indicate the stage of a predatory discussion. For instance, the usage of sexual words would likely indicate that the conversation had reached the sexual stage rather than being in the friendship forming or relationship forming stages. Understanding the structure of a predatory chat can prove valuable in the future for enhancing detection methods.

By employing linguistic-based analysis and leveraging word categories as predictors, this approach sheds light on the distinct stages of cyber grooming conversations. It provides insights into the proportional distribution of these stages and the potential to detect predatory discussions by analyzing language patterns and transitions between stages. Such knowledge can contribute to the development of more effective strategies for early detection and intervention in cases of cyber grooming.

2.2.2 Fine tuning BERT for text classification

In the paper titled "How to fine tune BERT for text classification, the authors explore various methods of how BERT can be effectively fine tuned for text classification tasks[SQXH19]. The paper begins with the authors highlighting the potential that BERT has when it comes to the classification of texts compared to other natural language processing models.They discuss methods including text preprocessing, architecture changes specific to the task at hand and optimization techniques.

The authors also outline the fune tuning process. This includes things such as the selection of data sets, tokenization strategies and the formatting of input. Hyper parameter settings are discussed as well as a part of this.

Towards the end of the paper, the authors present experimental results and analysis to present how effective a successful fine tuning approach can be. They discuss the influence of various factors, such as the size of the training set, learning rate, and batch size, on the model's performance. Additionally, the paper discusses strategies for handling imbalanced data sets and explores techniques to mitigate overfitting. Overall, this paper provides a comprehensive guide on how to fine-tune BERT for text classification.

2.2.3 Fine tuning BERT for cyberbullying detection

While not in the realm of cyber grooming work has been done on using BERT to detect other malicious behavioural patterns online, such as cyberbullying[YKC20]. Cyberbullying is the act of harming, intimidating, or harassing individuals through online platforms. In their work they used several different text corpuses containing regular conversations and cyber bullying. Each conversation had a binary value, it was either innocent or contained cyber bullying. The model was fed these conversations and marked them as one or the other. Their was a large imbalance between the number of conversations that were innocent and which contained instances of bullying. They chose to solve this by oversampling the bullying conversations so that the imbalance was not as large. This oversampling drastically improved the accuracy of the model.

2.2.4 Detection of cyber grooming

Various approaches have been made when trying to detect cyber grooming using natural language processing and machine learning. This paper focuses on three of them, message-based (MBD), author-based (ABD), and conversation-based (CBD) approaches[BK19].

In the message-based approach, only the words used in the chat are analyzed to determine if they were sent by a sexual predator. The author-based approach examines all messages sent by an individual participant in a chat to determine if they exhibit predatory behavior. The conversation-based approach assesses the entire chat and identifies suspicious conversations, designating one of the participants as the predator.

The performance of these approaches was evaluated using metrics such as recall, precision, and F1-score. . To extract features from each conversation, Bag of Words (BoW) and Term Frequency-Inverse Document Frequency (TF-IDF) techniques were utilized. These methods are commonly employed in Natural Language Processing and information retrieval to model and process text[Zho19]. TF-IDF, in combination with BoW, determines the importance of words in a text and assigns weights to signify their significance[Sco19]. The research employed three different classifiers: Logistic Regression (LogReg), Ridge, Naive Bayes, Support Vector Machine (SVM), and Neural Network (NN). These classifiers are used to categorize data into one or more classes. The study conducted tests using combinations of the different approaches (MBD, ABD, and CBD), feature sets, and classifiers.

The results demonstrated that instances of cyber grooming could be detected relatively early in the conversations. The conversation-based approach with a Neural Network classifier and the same approach with either Ridge or Naive Bayes classifiers yielded promising outcomes. In both cases, TF-IDF outperformed BoW as the preferred feature set. The findings highlight the effectiveness of the conversation-based approach and the significance of the selected classifiers and feature sets in detecting cyber grooming.

2.2.5 Detection of cyber grooming using BERT

Machine learning has emerged as a promising approach for detecting cyber grooming. Several research studies have been conducted to explore this field, yielding noteworthy outcomes. One study conducted focused on the use of BERT to achieve this. They compared the performance of three BERT versions (BERT-large, BERT-base, and mobileBERT) with other machine learning models. Surprisingly, BERT outperformed other at the time state-of-the-art models, including the resource-efficient mobileBERT.

To evaluate the effectiveness of BERT, the researchers employed a two-layer classification approach. Initially, the models received small portions of text and continuously classified them based on content. Following this, a second layer determined whether a grooming warning should be issued based on the collective classifications of the windows. The first layer assigned a "skepticism" value to each window, and if the sum of these values exceeded a predetermined threshold, it indicated the occurrence of cyber grooming in the analyzed chat.

These findings highlight the superiority of BERT over other models in the detection of cyber grooming. By utilizing a multi-layer classification approach and leveraging BERT's capabilities, the researchers achieved effective early detection of predatory chats. The study demonstrates the potential of BERT, showcasing its efficiency in addressing the challenges associated with cyber grooming detection.

2.3 State of the art

The detection of cyber grooming is a very important issue when it comes to safeguarding children on the internet. In order to help increase detection and aide the appropriate authorities machine learning and natural language processing can be used as a tool. My master thesis will focus on that issue, specifically with BERT, a transformer based machine learning model to solve this issue.

Cyber grooming predominantly occurs through written text communication, which often adopts a more informal and casual language. Additionally, online predators may intentionally employ expressions and vocabulary familiar to younger individuals to disguise their true age. This poses a challenge for BERT, as it was initially trained on formal language found in sources like Wikipedia and books[Tou19]. These texts typically lack slang, misspellings, and emojis, making it potentially difficult for BERT to analyze language containing these. This in turn may weaken BERT's ability to detect cyber grooming in conversations containing a large amount of such informal language.

To address this issue, my research will explore the fine-tuning of BERT to better comprehend commonly used abbreviations and other slang expressions. Existing studies have investigated these topics in the domains of detecting cyber grooming and utilizing BERT for sentiment analysis.

Several papers have examined the use of machine learning and neural networks for cyber grooming detection[BK19; VLA21]. They discuss various NLP models, including BERT, to detect predatory behavior in chats as early as possible. However, there is limited research exploring the impact of slang and abbreviations on cyber grooming detection. Fortunately, existing research demonstrates how BERT can be fine-tuned for better understanding of new language, sentiment analysis, and the detection of cyberbullying[TOYS20; SQXH19; YKC20]. While these studies conclude that BERT is effective„ the significance of abbreviation and slang usage remains unexplored. This thesis aims to fill this research gap.

Chapter 3 Data

3.1 Data collection and preparation

The data used during this thesis came from two sources, the PAN12 data set and from AIBA AS.

The data from the PAN12 set is open source and was made during an international competition for online identification of sexual predators. The data set is comprised of online one-to-one chats, some being predatory and others innocent. In order to differentiate predatory conversations from non-predatory ones, a supplementary text file provided alongside the dataset was utilized. This file contained a list of author IDs associated with predatory individuals. Consequently, any conversation containing any of these predator IDs was classified as predatory, while those without such IDs were considered non-predatory. Given the extensive scale of the database, the decision was made to exclusively employ the training set for the subsequent analysis. This approach was adopted with the intention of conserving computational resources and minimizing the computational time required for the analysis.

The data obtained from AIBA AS consisted of conversations derived from online video games meant for children. Unlike the PAN12 dataset, the predatory messages within the AIBA AS dataset were authentic predatory conversations. However, an major difference between the two datasets lies in the absence of any labels or markings for predatory conversations within the AIBA AS dataset. Consequently, devising an effective method to seperate innocent chats from predatory ones posed a great challenge. As a result, all conversations used from AIBA AS were manually examined, with each individual conversation being marked as either innocent or predatory.

This aspect of the manual examination presented a number of challenges, as my primary objective was to construct a BERT model specifically designed for detecting instances of cyber grooming. However, during this process several messages that may not have qualified as cyber grooming where encountered. This was due to multiple factors, such as the ages of the participants in the conversations were not mentioned, or the messages were not explicitly sexual. Nonetheless, considering the fact that these messages were exchanged on an online game intended for young children, their are no context in which such messages would be appropriate.

In the AIBA data set, each entry was an a data frame with several columns. This entries only contained a single message, and not a whole conversation. To fix this, each message had a unique conversation ID and timestamp of when it was sent. In order to analyze conversations and not individual messages, all messages where grouped togther by their conversation ID and placed in chronological order. After this had been done, each conversation was exported to a CSV file, where each row contained an entire conversation.

During the data cleaning process, no modifications were initially made to the messages. This decision was made since the goal for the research was to assess the performance of BERT in analyzing cyber grooming within internet chats, which commonly exhibit informal forms of English. Such informality encompasses the usage of abbreviations, emojis, and non-standard spelling variations of words. Therefore, preserving the original characteristics of the messages allowed for a more realistic evaluation of BERT's capability in handling the nuances and linguistic aspects prevalent in cyber grooming conversations.

Before the data was fed into a BERT model several steps are taken. These are:

- **Tokenization:** Each message in the data set was tokenized using the BERT tokenizer. The tokenizer changes the text into tokens that correspond with the vocabulary of BERT. This tokenization also add special tokens. '[CLS]' is added to the start of the text, and '[SEP]' is added at the end of each sentence. These special tokens are necessary for BERT to work as intended. A token is the smallest unit of text that BERT can understand. For example, the sentence "My name is Bob", could be tokenized into ["My","name", "is","Bob"].

- **Encoding:** When the text has been tokenized, they are encoded into input IDs. Each of the IDs correspond with a token in the vocabulary of BERT. During this process, padding and truncation is also used, which ensures that all of the sequences are the same length. Lastly an attention mask is introduced. This helps the model to avoid processing padded tokens, as they contain no relevant information.

- **Date conversion to PyTorch tensors:** The input IDs and attention masks are converted into PyTorch tensors. This are multi-dimensional matrices containing elements of a single data type. These are a fundamental data structure for PyTorch.

- **Data packaging:** When the tensors have been created, they are packaged into a PyTorch dataset. This dataset is an object which manages access to the data. An important element of this is the way allows the data to be loaded into batches through a data loader. In this context, a batch is a subset of the data set that is used for a single update of the model's weights during training. The use of batches is important because the data set in it's entirety is too large to be fed into the model.

- **Data Loading** A data loader is used to create a stream of batches during training and evaluation. It does this by handling processes such as sampling, shuffling and batch formation. Sampling is the process of selecting a subset of data from that represents the larger set. Shuffling is used to randomize the order of examples in the data set. This an important step as it helps to prevent any bias that may occur if the data is stays in it's original order. This means that the model well hopefully perform better on generalized data and does not overfit data. A shuffle occurs at the start of each new epoch.

  After these steps are finished, the data is ready to be consumed by the BERT model.

3.2 Data description

3.2.1 PAN12 data

The PAN12 data was originally stored in was stored in an XML-file. Each individual entry into the file was a conversation. This conversation had a unique ID and consisted of multiple messages. Each message consisted of several elements. These being message line(which order it is in the conversation), author ID, time stamp and text.

3.2.2 AIBA AS data

The AIBA as data was stored as parquet files. These files had multiple columns. These being:

- dateUTC: A time stamp of when the message was sent
- messageID: The unique ID of the message
- context: The unique ID of the chat the message is a part of
- gameID: Which game the message came from
- initiator: The unique ID of the person sending the message

  - reciever: The unique ID of the person recieving the message
  - content: The message being sent

This is the number of conversations that were used during my thesis. It is important to acknowledge that the PAN12 dataset substantially outweighs the data collected from AIBA AS in terms of volume. This was kept in mind when analyzing the results achieved from both models when trained on their respective data.

	AIBA AS	PAN12
Innocent conversations	2324	66927
Predatory conversations	135	2016
Total conversations	2459	68943
Ratio predatory to innocent	5.8%	3.0%

3.3 Statistics of PAN12 conversation data

In total 68942 conversations make up the PAN12 data set. These graphs and tables give some insights into the structure and length of these conversations for the data set as a whole, but also for the innocent and predatory conversations.

Figure 3.1: Graph of messages in all PAN12 conversations

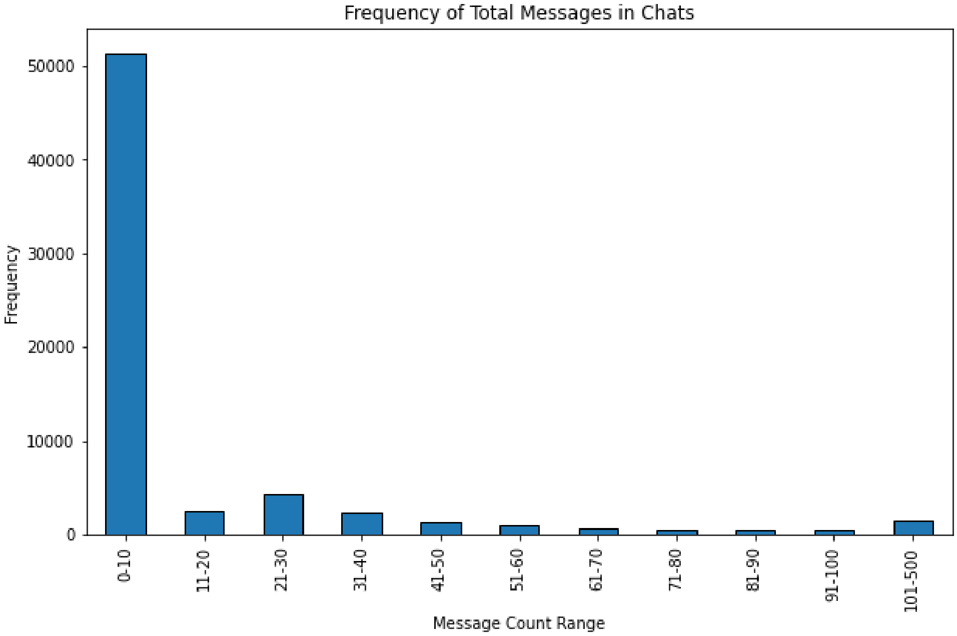

Table 3.1: Statistics of all messages in PAN 12 conversations

Average number of messages in conversations	13.5
Median number of messages in conversations	4

Figure 3.2: Graph of messages in innocent PAN12 conversations

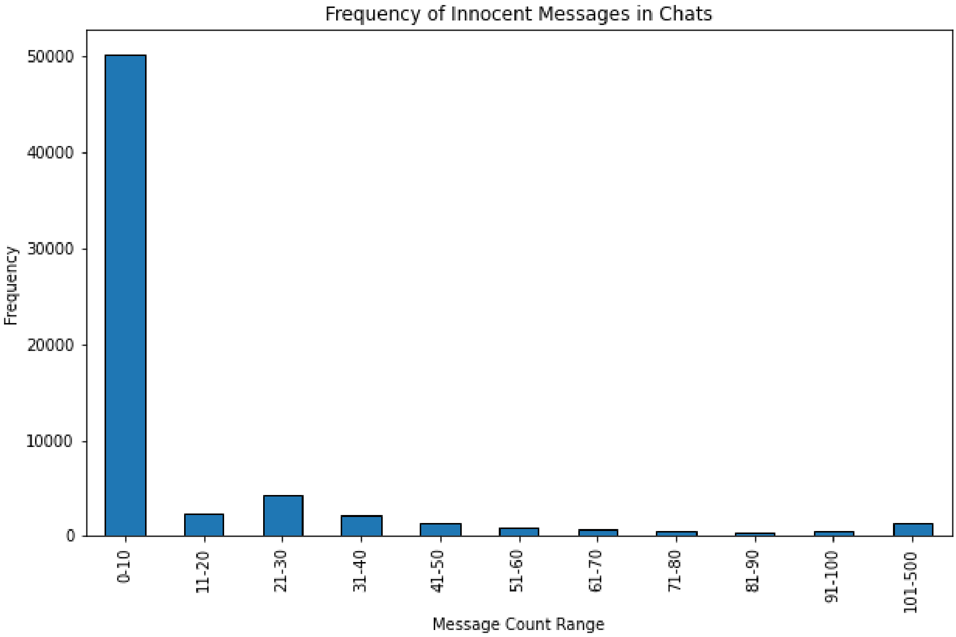

Table 3.2: Statistics of innocent messages in PAN 12 conversations

Average number of messages in conversations	12.7
Median number of messages in conversations	4

Figure 3.3: Graph of messages in predatory PAN12 conversations

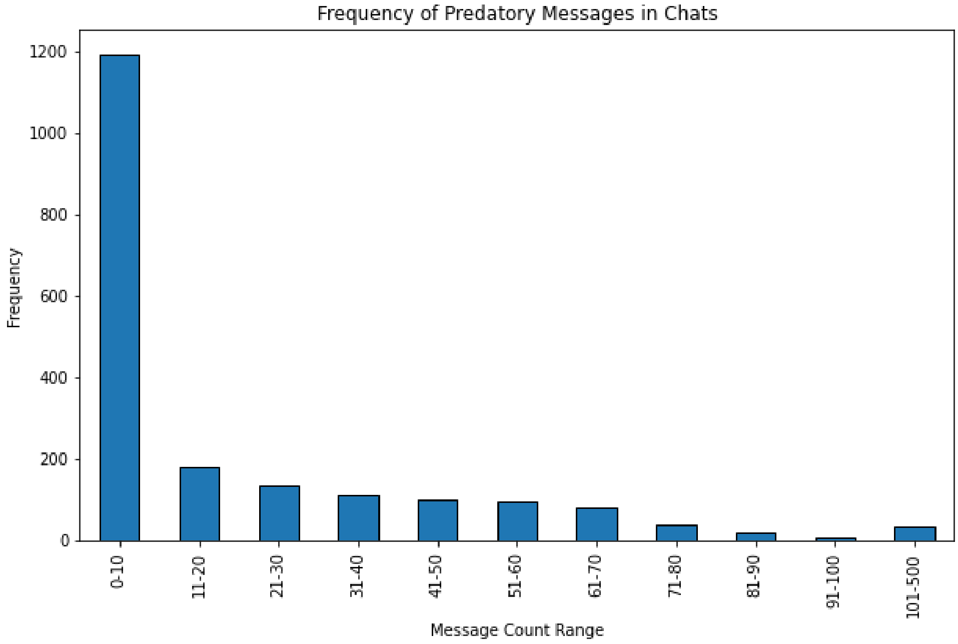

Table 3.3: Statistics of predatory messages in PAN 12 conversations

Average number of messages in conversations	20.3
Median number of messages in conversations	4

3.4 Statistics of AIBA AS conversation data

In total 2459 conversations make up the AIBA AS data set used in this thesis. These graphs and tables give some insights into the structure and length of these conversations for the data set as a whole, but also for the innocent and predatory conversations.

This graph shows the frequency of the total messages in AIBA AS each conversation.

Figure 3.4: Graph of messages in all AIBA AS conversations

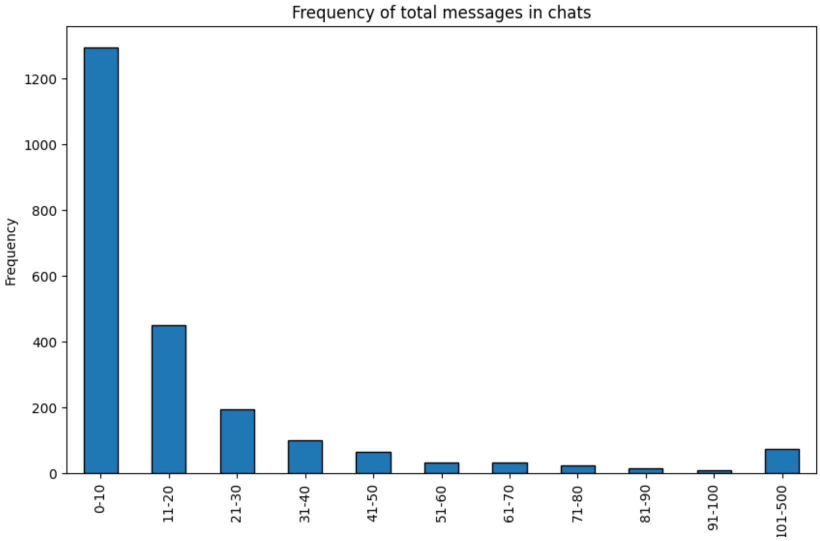

Table 3.4: Statistics of messages in all AIBA AS conversations

Average number of messages in conversations	34.6
Median number of messages in conversations	8

This graph shows the frequency of the total messages in each innocent conversation.

Figure 3.5: Graph of messages in innocent AIBA AS conversations

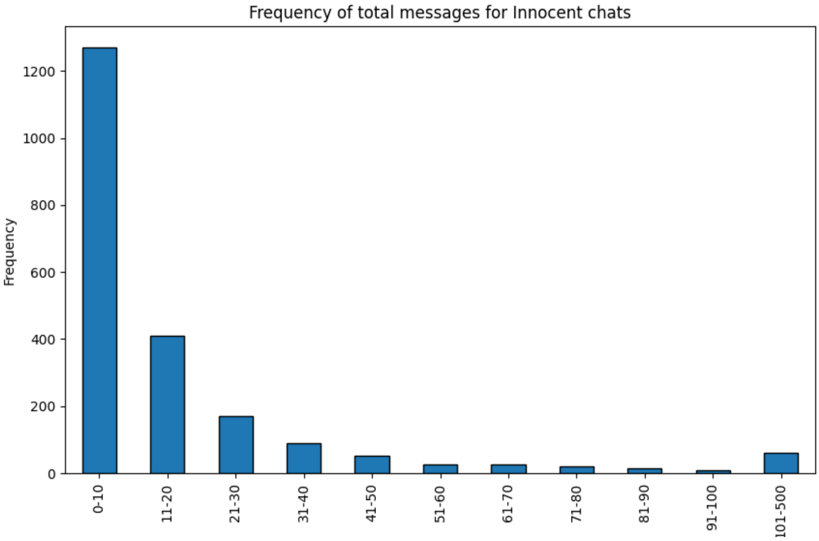

Table 3.5: Statistics of messages in innocent AIBA AS conversations

Average number of messages in innocent conversations	31.4
Median number of messages in innocent conversations	7

This graph shows the frequency of the total messages in each AIBA As predatory conversation.

Figure 3.6: Graph of messages in predatory AIBA AS conversations

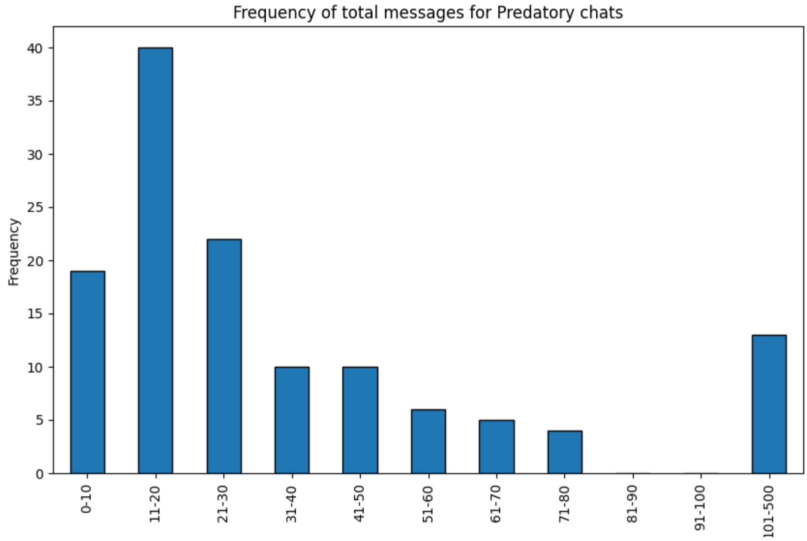

Table 3.6: Statistics of messages in predatory AIBA ASconversations

Average number of messages in predatory conversations	85.9
Median number of messages in predatory conversations	23

The analysis of the data extracted from the AIBA AS and PAN12 conversations give some insights into the overall characteristics of the chats. A notable observation is that a significant portion of the chats in both datasets are relatively short, consisting of less than 10 messages.

It is worth noting that there is a considerable disparity between the median and average number of messages per conversation, indicating a right-skewed distribution within the dataset. This implies that the majority of conversations exhibit a smaller number of messages, while a few conversations contain a notably larger number of messages.

This distribution pattern can potentially be attributed to various factors. For instance, it could be indicative of some conversations being particularly active or spanning over extended periods of time, thus accumulating a higher volume of messages compared to others. These factors contribute to the skewness of the dataset and should be taken into account when interpreting the results and drawing conclusions based on the analysis.

Another interesting finding is that predatory conversations in the AIBA AS data tend to have a higher minimum number of messages compared to innocent conversations. This means that, on average, predatory conversations involve more messages than innocent ones. This distinction could serve as a useful indicator for BERT in distinguishing between the two types of conversations. By considering the minimum message count as a distinguishing factor, BERT may be better equipped to identify predatory chats and differentiate them from innocent ones. This insight can potentially enhance the effectiveness of the BERT model in accurately detecting and classifying predatory behavior in online conversations.

3.5 Data with abbreviation replacement

A key objective of my thesis was to investigate the impact of abbreviations on BERT's ability to detect cyber grooming. To address this challenge, I employed a specific approach: training and testing BERT on two identical datasets, with the only difference being that one dataset underwent a modification process. This modification involved replacing a significant portion of abbreviations with their original expanded forms in formal English. To accomplish this, a Python script was utilized to automatically process all conversations in both the training and test datasets.

The selection of abbreviations to replace was determined through a careful process. Firstly, approximately 100 unique conversations, each consisting of over eight messages, were carefully reviewed. Additionally, a list of commonly used abbreviations sourced from a student also conducting research on the AIBA AS data set was used. The most frequently encountered abbreviations were identified and subsequently targeted for replacement. In total, 77 unique abbreviations were replaced throughout the datasets. The complete list of the replaced abbreviations can be found in the appendix.

By using this replacement approach, my research aimed to assess how BERT's performance in detecting cyber grooming would be influenced by replacing abbreviations with their expanded form. This analysis will provide valuable insights into the role of abbreviations in BERT's understanding and interpretation of online conversations related to cyber grooming.

Regrettably, the table provided does not include many abbreviations explicitly related to sexual or predatory behaviour. While manually reviewing the messages in the AIBA AS data set, I did encounter several instances of such abbreviations; however, many of them were spelled differently. This can be attributed, in part, to the moderation and censoring mechanisms implemented by the online games these chats originated from. The game restricts the usage of certain words, particularly profanities. This leads individuals to use creative and unique alternative spelling methods. Many of these unique spellings were specific to individual chatters, making it challenging to establish an efficient method for detecting and replacing them.

Due to the inherent complexity and variability of these unique and creatively spelled abbreviations, finding a comprehensive solution for their detection and replacement proved to be impractical within the scope of this research.

The number of times each of these abbreviations appeared in both the AIBA AS and PAN12 data sets can aslo be viewed in the appendix. When analyzing the data for the usage of abbreviations, several trends emerge. One of the most evident patterns is that the AIBA AS dataset shows a significantly higher frequency of abbreviations compared to the PAN12 dataset. This is quite noteworthy considering that the PAN12 dataset encompasses a larger volume of 68,943 unique conversations, whereas the AIBA AS dataset comprises only 2,459 conversations. Despite this difference in amount of conversations, the AIBA AS dataset contains a greater number of abbreviations. This difference could imply that the nature of conversations within the PAN12 dataset tends to be more formal. It is possible that factors such as demographics or the targeted audience contribute to this distinction. The conversations from the AIBA AS dataset originate from a children's game, suggesting that the age range of the individuals engaging in the chats is younger in comparison to the individuals involved in the PAN12 chats.

The usage of abbreviations within the AIBA AS dataset is prevalent in both predatory and innocent conversations. Among these abbreviations, one of the most frequently used is "u," which represents the word "you." An interesting observation is that "u" appears more frequently in predatory conversations compared to innocent ones. Specifically, it is utilized approximately 8.33 times per predatory conversation, whereas in innocent conversations, it appears around 2.13 times. This discrepancy suggests that in predatory conversations, there is a higher occurrence of direct addressing towards the other person.

However, it is important to note that these trends need to be considered alongside other indicators in order to help determine if a ongoing conversation is predatory or innocent.

3.6 Data quality and limitations

There are some issues with the way the PAN2012 data set is structured. Due to the data being split into smaller unordered segments, it is not possible to test on longer and continuous chats. This is problematic because cyber grooming often takes place over time where the predator builds trust and a relationship with their victims. This means that the entire grooming process may not be included in a singular segment. The choice of segmenting predatory chats into smaller parts could also serve as en explanation of why the number of shorter conversations vastly outnumber the longer ones.

Another issue is the use of volunteers. Since they were all adults posing as children, their answers may therefor not be representative of what an actual child would answer[18]. This is not an issue for the conversations from the AIBA AS data. As The predatory conversations in that data set include instances of people trying to groom minors.

During the process of gathering enough chats, the creators of the PAN2012 data set found it hard to find enough chats with people having "regular" conversations about everyday topics such as politics, current events or pop culture. This led them to using chats from IRC channels, which mostly discuss quite technical and uncommon subjects. It could be argued that the data should include more common place and ordinary conversations. As for the AIBA AS conversations, most of them revolve around the online game from which all of the chat logs are retrieved from. In the context of detecting cyber grooming on those sorts of platforms, the importance of a variety of different conversation topics may not be that important. How ever if the goal is to develop BERT to be able to detect cyber grooming on a more general basis, chats discussing more diverse subjects could be implemented in the data set.

An issue that arose when gathering data from the AIBA As data set was that I was not able to find enough data that could be deemed as grooming. In many of the conversations the ages of the participants where never explicitly mentioned. How ever due to the context of that these chats were taking place on a platform targeted for children, they were flagged as predatory because such conversations should not be taking place with minors involved. This justification was also taken when flagging other conversations as grooming or predatory. This included instances of sexual role play and sexting.

3.7 Ethical considerations

Due to the sensitive nature of the data it is important that it is treated carefully. This is the case for both the AIBA AS and PAN12 data even though they are slightly different.

The predatory chats present in the PAN12 papers are not from actual instances of cyber grooming, but from adults pretending to be children in order to lure sexual predators in the hope of exposing them. These conversations are all public on the website of the American organization Perverted Justice. While all mentions of names, telephone numbers, emails and so on have been censored in the PAN12 data set I have used, the chat logs can be traced through the website of Perverted Justice. Though this web site the personal details are of the predators are available for the public to see.

The data from AIBA AS on the other hand, is from actual conversations between sexual predators and children. Naturally this makes them far more sensitive than the PAN12 data, thus it not being publicly available. Sharing this data set outside of AIBA AS own specifies digital workspaces could lead to legal consequences from authorities as minors are being groomed and potentially harmed in these chats. Another major difference between the AIBA AS data and PAN12 data that I noticed when reading, was that the AIBA AS data contained more personal information related to social media accounts such as snapchat, discord and instagram. This could potentially be seen as a violation of privacy.

3.8 Summary and Conclusion

The data derived from the AIBA AS and PAN12 have many similarities and difference. One of the largest similarities between them is that the majority of the conversations are under 10 messages long. That could pose some problems as grooming traditionally happens over a longer amount of time, which means that this process may not be accurately portrayed in such a small amount of messages.

A major difference between the two data sets is that the chats from AIBA AS tends to contain much more informal language than those from PAN12. This suspicion arose when several PAN12 conversations were sampled and read manually to be compared to messages in the AIBA AS. This notion was strengthened when conducting the abbreviation analysis on both data sets. The chats in the AIBA AS data contained a greater amount of abbrevaitions, despite being much smaller than the PAN12 data set.

Due to the sensitive nature of the data sets, I was not able to find any other credible sources of data that could be used for training a BERT model for cyber grooming detection. When considering the rarity of chat logs related to cyber grooming, the AIBA AS data sets offers a unique opportunity to conduct more research into this field, especially on what type of effect informal language and internet slang has on machine learning models abilities to detect cyber grooming.

Chapter 4 Base analysis

This chapter will be discussing how the original BERT models where developed and trained for the detection of cyber grooming.

In this stage of the research, three distinct working models were trained to explore and measure the performance variations of BERT depending on the data set. The underlying code used for developing these BERT base models remained the same, while the datasets employed for training them differed.

The first model was trained using the complete PAN12 dataset, containing the entirety of its conversations. The second model utilized the AIBA AS dataset, consisting of all conversations manaully labeled prior. Lastly, a modified version of the PAN12 dataset was created to match the number of innocent and predatory conversations present in the AIBA AS dataset, and this dataset was used to train the third model.

The objective behind employing these different datasets was to evaluate and compare how BERT's performance differed when applied to chats used by AIBA AS versus the PAN12 dataset. By training the models on these distinct datasets, the research aimed to examine the impact these differences had on BERT's detection and classification performance of cyber grooming. This approach enabled an analysis of BERT's performance across the different datasets, shedding light on the strengths and weaknesses of the model when applied to various chat sources.

The decision to utilize BERT base for the initial analysis was based on several factors. One crucial factor is that BERT base is computationally less demanding compared to BERT large[Tou19]. This is mainly because BERT base has fewer parameters, allowing for quicker training. This makes BERT base an ideal choice for the initial analysis. By establishing the performance of BERT base as a baseline, it becomes relatively straightforward to fine-tune the model by adjusting its parameters or architecture[Tou19].

4.1 Description of code

The code used to train the BERT model can be viewed in the appendix. The following subchapters describe how it is implemented

4.1.1 Data preparation

In this step the raw text is translated into a numerical form that can be understood by BERT. This includes tokens, attention masks, and token type IDs. This, and the initial loading of the CSV file containing the conversations, are all handled in the **ChatDataset** class.

4.1.2 Model creation

The **BertForSequenceClassification** model is in this case a BERT base uncased variant. It also contains an additional layer which is used for sequence classification. The argument **num Labels** specifies that the classification that the BERT model will be doing is binary, as the value is set to 2. This means that the conversations to be analyzed will either be predatory or innocent.

4.1.3 Model training

The training process for the model uses an AdamW optimizer, which is a variant of the Adam optimization algorithm. Adam stands for Adaptive Moment Estimation. The choice of using AdamW over stand Adam was that it is often the better alternative when the use case is fine tuning an existing model, which is the case here[LH19].

A learning rate scheduler with a warm up period is also used during this stage. This controls how much the parameters of the model are adjusted at each step of the training process. This allows the learning rate to change over time. This typically starts at a low value and increases over time. Towards the end of the training process it decreases again. The model will then make large updates to it's parameters early in the training, before adjusting them more precisely as the training continues and the model has a better understanding of the data.

4.1.4 Evaluation and prediction during training

This process happens in parallel with the training process. The evaluation is conducted through testing the model on unseen data. This will show an indication of how well the model is able to generalize. This is important, as a model that only performs well on the data it has been trained on is not useful in practice. No parameters are updated during this process.

When the model has been trained and evaluated, the next stage involves having it make predictions on unseen data. The function **getPredictions** in the code does this. The model is put into evaluation mode so that none of the parameters are updated. These predictions take places by passing inputs through the model and receiving outputs. These outputs are the interpreted as the predictions. More specifically, these are "logits" for each class. This are raw and unnormalized scores. The class with the highest score is selected to be the models prediction, this happens through the **torch.argmax** function. These predicitons and scores are saved for later as they are used for calculating performance metrics.

4.1.5 Performance metrics

Performance metrics are used to measure how well the model is performing. For this process, three metrics were used, precision, recall and F1-score.

Precision: A measure of how many of the positive predictions are correct. Calculated by

$$\frac{TruePostives}{TruePositives + FalsePositives} \tag{4.1}$$

Recall: A measure of how many of the positives cases are correctly predicted by the classifier. Calculated by

$$\frac{TruePostives}{TruePositives + FalseNegatives} \tag{4.2}$$

F1-score: Measured by combining both precision and recall.

$$\frac{2 * Precision * Recall}{Precision + Recall} \tag{4.3}$$

These values are measured after every epoch. If a better F1-score is registered than that done previously, the BERT model overwrites already existing model and is saved instead.

4.2 Performance of first BERT models

This table shows the performance of the first three BERT models that were trained. It is clear, that the model trained on the entire PAN12 data set outperforms the other two, but that is to be expected when the data set it was trained on was much larger than the other two. Another thing that is worth noting, is that the performance of the models between the AIBA AS data and the smaller PAN12 set was quite similar. It may seem that BERT does not struggle to understand the AIBA AS set

Table 4.1: Performance of initial BERT base models

	Entire PAN12	PAN12(Same size as AIBA AS date set)	AIBA AS
Precision	0.88	0.75	0.83
Recall	0.84	0.54	0.50
F1-score	0.86	0.63	0.62

compared to the PAN12 set, despite the messages in the AIBA AS set being much more informal, as evident in the table containing the number of abbreviations across the two data sets.

Epoch	Train Loss	Val Loss	Precision	Recall	F1
1	0.4127	0.2192	0.0000	0.0000	0.0000
2	0.2017	0.2014	0.0000	0.0000	0.0000
3	0.1656	0.1722	0.0000	0.0000	0.0000
4	0.1257	0.1583	0.6250	0.1786	0.2778
5	0.0818	0.1756	0.5769	0.5357	0.5556
6	0.0380	0.1854	0.4571	0.5714	0.5079
7	0.0165	0.1863	0.8235	0.5000	0.6222
8	0.0117	0.2104	0.7692	0.3571	0.4878
9	0.0086	0.2033	0.7500	0.4286	0.5455
10	0.0085	0.2023	0.6667	0.4286	0.5217

Table 4.2: Model trained on AIBA data

Epoch	Train Loss	Val Loss	Precision	Recall	F1
1	0.3569	0.1411	0.0000	0.0000	0.0000
2	0.1440	0.1332	0.4074	0.5000	0.4490
3	0.0953	0.1127	0.5500	0.5000	0.5238
4	0.0795	0.1347	1.0000	0.2273	0.3704
5	0.0582	0.1231	0.7059	0.5455	0.6154
6	0.0495	0.1287	0.7500	0.5455	0.6316
7	0.0433	0.1383	0.7500	0.5455	0.6316
8	0.0408	0.1382	0.7500	0.5455	0.6316
9	0.0367	0.1451	0.7500	0.5455	0.6316
10	0.0378	0.1464	0.7500	0.5455	0.6316

Table 4.3: Model trained on PAN12 data

4.3 Training results

These two tables show the results from training the similar sized AIBA AS and PAN12 data sets on two BERT base models. At the end of each epoch, the performance of the models were recorded. For the model trained on the AIBA AS data, the training loss was first measured to be 0.4127. Throughout each epoch it gradually decreased to 0.0085. This indicated that the model was learning from the training and was getting better at predicting which conversations were predatory or not.

In contrast to this however, the validation loss initially was reducing until the fourth epoch. It started at 0.2192, decreased to 0.1583, to then increase to 0.2023 for the last epoch. This could mean that the model was beginning to overfit the training data, meaning it would not generalize well on unseen data. This same trends could also be seen for the model that was trained on the PAN12 data.

Across all epochs in both training sets, the clear trend was that there was a decrease in loss and an increase performance metrics. This suggests that both models were learning effectively. In the PAN12 data, the F1-score increased or stayed the same over all of the epochs. This was not the case for the AIBA AS model. During it's training there was a noticeable fluctuation between the F1-scores. After some epochs it would increase, while also decrease. This could indicate that overfitting or unstable learning was taking place. This reinforces the importance of further refining the model training process to ensure robust and stable performance across different data sets.

4.4 Conclusion of base analysis

This training process gave some insights into the behaviour of BERT base when it came to training it for the detection of cyber grooming. The model trained on the PAN12 data set displayed a consistent improvement in it's performance metrics. On the other hand, the model trained on the AIBA AS data set showed a fluctuation in performance. This indicates that it may have to be further fine tuned. This could include adjusting some parameters, changing the architecture or implementing measures in order to prevent overfitting.

Chapter 5
Improvements upon performance

This chapter will be focusing on how the results from achieved in the previous chapter can be improved through fine tuning the initial BERT base model.

5.1 Comparing performance to other BERT versions

Even though BERT is a relatively new piece of technology(first introduced in 2018), several different models have been developed and tested in order to further enhance the original BERT models performances on specific tasks. These have taken initial model as a reference point, and made changes to things such as parameters and architecture. In this experiment, three different BERT models were testing to see if any improvements could be made upon the performance of the original BERT base model. These were RoBERTa, DistilBERT and ALBERT. These new models used the exact same code and default parameters as the initial BERT base model. The only difference between them was which of these model were loaded in to begin the training.

5.1.1 RoBERTa

RoBERTa stands for Robustly Optimized BERT approch. It is a BERT model which was developed by FaceBook AI. The major difference between it and it's predecessor BERT is in how it was trained. It uses the exact same architecture, but makes some changes in the pre-training process. RoBERTa applies dynamic masking rather than BERT's static masking. This means that the masked langauge model may be different for every epoch in training, which is not the case for BERT. This provides the model with a more diverse learning scenario. RoBERTa also utilizes a higher learning rate and larger batch sizes. The transformer based architecture in both RoBERTa and BERT is able to capture the dependencies between among words and sentences. However in RoBERTa, there is no NSP(Next sentence prediction) task. The developers purposefully chose to omit it[LOG+19].

5.1.2 DistilBERT

DistilBERT was developed by the Hugging Face team. It is basically a distilled or simplified version of BERT. It's aim is to be a smaller. faster and lighter model. It does this by removing several parameters and simplifying the architecture. Even though it has implemented these changes, it is still able to achieve a comparable performance to that of BERT. It is able to retain 95% of it's performance despite being 60% smaller and 60% faster. Similarly to BERT, DistilBERT is based on a transformer-layer architecture, but it has half the number of layers as BERT base. Despite this, the core structure of DestilBERT remains the same as BERT base, as it has both self attention mechanisms and feed forwarding networks. Similarly to RoBERTA, DistilBERT does not have any next sentence predictions in pre training. This is due to the fact that it removes token type embeddings[SDCW20].

5.1.3 ALBERT

ALBERT is short for A Lite BERT. Just like BERT, it was developed by Google. The main difference between the two is that ALBERT does not have as many parameters as BERT. This means that the model size is smaller, and that it theoretically will be faster to train. While doing this, it does not decrease the performance of the model. This is due to it factorizing the original embedding matrix into two smaller ones. This makes it easier to scale with out increasing the computational costs. The architectural differences between the two models is that ALBERT uses cross layer parameter sharing. This means that parameters are shared across all layers of the model. This in theory, reduces the size of the model. This also goes for the self attention mechanisms. It is shared across all the layers, while this is not the case in BERT base. Like both RoBERTa and DistilBERT, next sentence predictions is removed from pre training. Instead ALBERT uses Sentence Order Predictions(OSP)to bettee understand the context and order of sentences. OVerall,ALBERT allows for training on larger datasets and better catches long term dependencies[LCG+20].

5.1.4 Discussion of performance

The RoBERTa model started with a relatively high training and validation loss compared to the other models, however these values decreased considerably throughout training. It's performance were not relatively accurate until after five epochs. After that point, the performance metrics began to improve as the training went on as well. It achieved it's highest F1-score after the tenth and final epoch. Unfortunately, their was some fluctuation of the F1-score similar to what happened when training the BERT base model. This could be an indication of overfitting and that the model will generalize poorly. This can also be seen in the increase of validation loss.

Epoch	Train Loss	Val Loss	Precision	Recall	F1
1/10	0.3299	0.2491	0.0000	0.0000	0.0000
2/10	0.2012	0.1759	0.0000	0.0000	0.0000
3/10	0.1735	0.1602	0.0000	0.0000	0.0000
4/10	0.1356	0.1513	0.0000	0.0000	0.0000
5/10	0.0872	0.1498	0.4839	0.5357	0.5085
6/10	0.0574	0.1729	0.6500	0.4643	0.5417
7/10	0.0307	0.2247	0.6000	0.5357	0.5660
8/10	0.0199	0.2287	0.6111	0.3929	0.4783
9/10	0.0096	0.2373	0.6316	0.4286	0.5106
10/10	0.0120	0.2256	0.6154	0.5714	0.5926

Table 5.1: RoBERTa model trained on AIBA AS data

Epoch	Train Loss	Val Loss	Precision	Recall	F1
1/10	0.3944	0.2147	0.0000	0.0000	0.0000
2/10	0.1937	0.1819	0.0000	0.0000	0.0000
3/10	0.1530	0.1610	0.6667	0.0714	0.1290
4/10	0.0954	0.1681	0.4516	0.5000	0.4746
5/10	0.0648	0.2155	0.4667	0.2500	0.3256
6/10	0.0261	0.2316	0.4615	0.4286	0.4444
7/10	0.0142	0.2763	0.5714	0.2857	0.3810
8/10	0.0093	0.2641	0.5500	0.3929	0.4583
9/10	0.0064	0.2786	0.5882	0.3571	0.4444
10/10	0.0050	0.2828	0.5882	0.3571	0.4444

Table 5.2: DistilBERT model trained on AIBA AS data

DistilBERT had a relatively high training loss to begin with, but it reduced relatively quickly to achieve a low validation loss compared to the other models. Unlike the other models, DistilBERT did not perform well based on the metrics of precision, recall and F1-score. Based on the increase of the validation loss after the fifth epoch, overfitting could have been an issue in the training of the model.

ALBERT began the training with the lowest validation and training loss, showing that it was able to learn relatively quickly compared to the other models. Although this score is lower than RoBERTa's, it surpassed DistilBERT's result, showcasing a stronger balance between precision and recall. However, similar to the other models, ALBERT demonstrated signs of overfitting from the 5th epoch onward, with

Epoch	Train Loss	Val Loss	Precision	Recall	F1
1/10	0.2468	0.1941	0.0000	0.0000	0.0000
2/10	0.1839	0.1931	0.0000	0.0000	0.0000
3/10	0.1744	0.1851	0.0000	0.0000	0.0000
4/10	0.1430	0.1504	0.6500	0.4643	0.5417
5/10	0.0992	0.1737	0.7778	0.2500	0.3784
6/10	0.0742	0.1908	0.7500	0.3214	0.4500
7/10	0.0425	0.2316	0.6667	0.2143	0.3243
8/10	0.0143	0.2295	0.5714	0.4286	0.4898
9/10	0.0073	0.2427	0.6000	0.4286	0.5000
10/10	0.0051	0.2505	0.6316	0.4286	0.5106

Table 5.3: ALBERT model trained on AIBA AS data

validation loss increasing.

When compared the performance of these three models to the BERT base model that was initially trained, several patterns emerge. RoBERTa achieved the best results followed by ALBERT and DestilBERT. BERT base still outperformed these models, but the overall performance of RoBERTa and BERT base was comparable to each other. The difference between the F1-scores being only being around three percentage points. All models, including BERT base, demonstrate a tendency toward overfitting after a certain number of epochs, suggesting that all models could benefit from techniques like early stopping, regularization, or dropout to improve generalization on unseen data.

5.2 Effect of abbreviaitons on BERT's performance

Another focus of of this thesis was to see how the usage of emojis, abbreviations and other informal language forms effect BERT's ability to detect cyber grooming in chats. When looking through the messages in the AIBA AS data set, the usage of emojis was unfortunately not as common as the usage of abbreviaitons. So much so, that I believed that their importance in BERT's detection of cyber grooming may have been irrelevant, especially when compared to abbreviations. The few emojis that were discovered in the conversations where variations of a smiling or winking face. Abbreviations on the other hand, were more varied in their usage and forms.

In order to test how much BERT's performance was effected by the use of abbreviations in the detection of cyber grooming, two different models where trained using two slightly different data sets. These would be identical but with a slight difference. One of them contained the original AIBA AS conversations, while the other set contained the same ones, except that many of the abbreviations and slang expressions were replaced with their original expansions and formal forms. Take for example this sentence, "HBU? How r u doing". This would be translated to "How about you? How are you doing". This process was implemented through a python script for both the test and training sets. Then two different BERT base models would be trained using the two different training sets. Their performance would then be measured on the test sets containing conversations with abbreviaitons, and conversations only containing their respective expansions. Similar to earlier tests, the performance of the models were measured after each epoch using precision, recall and F1-score. After all ten epochs were completed, the one model with the best F1-score was saved and used for more testing. These tables contain the training results from this process. Similar to the other models previously trained, their was a fluctuation in the F1-scores and an increase in validation results, indicating overfitting.

Once these models had been trained, their performance was measured using both test sets. This was done in order to see how the models would react to data where many of the abbreviations would either be present or absent. The model trained on the original conversations is named Original model, and the one trained on the conversations with many abbreviations replaced is called Expansion model. How they scored when tested can be viewed in the tables below:

Model	Precision	Recall	F1-Score
Original model	0.75	0.54	0.63
Expansion model	0.47	0.29	0.36

Table 5.4: Performance comparison of models on original data

Model	Precision	Recall	F1-Score
Original model	0.83	0.36	0.50
Expansion model	0.52	0.46	0.49

Table 5.5: Performance comparison of models on data with abbreviation replacement.

5.2.1 Discussion of results

When examaning the results in the tables, some trends can be spotted. Firstly, the model trained on the original data set from AIBA AS(with abbreviations and other internet slang present), it is clear that it outperforms the expansion model on all metrics. It is able to detect true positives at a much higher rate, while also minimizing the odds of detecting any false positives and negatives. These results suggest that incorporating abbreviations and internet slang replacements directly into the training set does not enhance performance if the model has been trained on more informal language.

When looking at the results from the model trained on the data were abbreviations and slang has been replacced, the opposite occurs. It is better able to detect cyber grooming in the conversations with more formal language. This however could be expected, as the model does not have as large a basis as the previous model to understand more informal language, as much of it has been replaced during the training phase. An interesting result that occured is that the model trained on the the original conversations maintained a high precision score, but the recall dropped significantly when compared to the previous model. This would suggest it struggled to correctly identify true positives when faced with a more formal language. However when it first concluded that a conversation was predatory, the chance of it being so was high.

The model trained with abbreviations and slang seems to perform better when tested on data that resembles its training set. However, it underperforms when faced with data outside of its training scope (i.e., data without abbreviations). A result from this process that is worth noting is the scores from each individual model when tested on the test sets corresponding to the language form they were trained on. One might expect that since BERT has been trained on more traditional versions of English, it would be able to detect cyber grooming better in the conversations that were more formal, but that did not seem to be the case. The model which was trained on the original data measured an F1-score of 0.63 when tested on a test set of the original data. The model that was trained on conversations where the many of the abbreviations were replaced scored 0.53. This is a difference of 14% which is quite significant.

The reasons for this could be numerous. It may be plausible that the model trained on the original data learned to recognize certain underlying patterns or structures in the text that are independent of the abbreviations or internet slang. These features could include sentence structure, phraseology or other contextual features that are not directly tied to the presence of abbreviations but are still indicative of cyber grooming behavior.

Unfortunately not all of the abbreviations and internet slang in the conversations where able to be replaced. It is difficult to conclude how many were overlooked, but one thing that is certain is that a large number of the sexual and predatory words were not able to be replaced. This was due to the fact that they were spelled and represented in numerous different ways. This made it difficult to develop a method that would efficiently replace all of them. It would be interesting to compare the results if such a method used to change most of the abbreviations and other informal language forms in these conversations.

However, these results could seem to indicate that replacing commonly used abbreviations and slang in online conversations with their formal expansions does not automatically increase BERTs ability to detect cyber grooming. This implies that despite these conversations being quite different from the majority of the language that BERT has been trained on(Wikipedia articles and books[Tou19]), it's understanding of English was good enough that it was able to comprehend the language quite well.

Something worth exploring is if any of the abbreviations are contained inside of BERT's vocabulary. BERT has a fixed vocabulary that uses WordPiece tokenization. This breaks words into smaller subwords if the word is not in its vocabulary. This way, it can handle almost any word it encounters, even if it wasn't in the original vocabulary[Tou19]. When checking the vocabulary of BERT base too see if they contained any of the abbreviations, some of them where not in it. These were:

wby, hbu, rp, idc, yk, hru, btw, wyd, brb, idk, alr, tbh, lol, ofc, plz, afk, wtf, pics, tysm, b4, rlly, irl, jk, lemme, ngl, thx, smth, lgtm, nvm, gf, bby, wth, inv, txt, gimme, wym, lmao, smh, ight, xo, x o, x-o, m8, ily, ppl, yolo, ttyl, sup, yh

All of the other abbreviations where contained in BERT base's tokenizer. To see how important the presence of these abbreviations in BERT's vocabulary is, two different models where trained. One a training set where only the abbreviations in the vocabulary was replaced, and one where only the abbreviations that were not in the vocabulary was replaced. Their performance was then measured on the same test sets of the previous models. One with the original AIBA conversations, and one were many of the abbreviations were replaced. These were the results:

Model	**Precision**	**Recall**	**F1-Score**
Model trained with abbreviations in tokenizer replaced	0.69	0.32	0.44
Model trained without abbreviations in tokenizer replaced	0.83	0.36	0.50

Table 5.6: Performance comparison of models on original data.

When looking at these results some interesting trends appear. While the F1-scores of the models where as a whole not larger than the ones trained previously, the precision scores in the model trained without the abbreviations in the tokenizers present in the training set, scored higher on precision. The number of predatory conversations it marked as predatory where indeed predatory 86% of the time. How ever due to the low recall score it showed a poor ability to correctly identify all of the

Model	Precision	Recall	F1-Score
Model trained with abbreviations in tokenizer replaced	0.70	0.50	0.58
Model trained without abbreviations in tokenizer replaced	0.86	0.21	0.34

Table 5.7: Performance comparison of models on data with abbreviation replacement.

predatory conversations in the test set, only being able to detect 20%. If precision is the only metric to measure the performance of the model, it would score quite well, however due to the very low recall rate, the total performance of the models trained with abbreviations replacement based on their inclusion in BERTs vocabulary does not achieve a better result than the inital BERT model trained.

From these results it would seem that BERT possesses the ability to understand text relatively well despite it containing slang and abbreviations. Replacing and changing this language has little to no effect. To the contrary, it seems to decrease the overall performance of the model. The only exception seems to be when it comes to the precision of the models, but this will also result in a poor recall score in comparison.

From these results it could be expected that the BERT model trained on the AIBA AS conversations can perform similarly to the model trained on the entire PAN12 data set, given enough data. This can be seen in the performance of the models that were tested on similarly sized data sets. If their performance on a smaller data set was similar, it could be speculated that if the AIBA AS data set increased in size similar to that of the PAN12 set, it too would be able to perform similarly. That would mean measuring precision, recall and F1-scores over 80%, despite the conversations from AIBA AS being more informal overall.

For these results to be more conclusive, ideally all or most instances of internet slang and abbreviations should be replaced in AIBA AS data sets. Even after replacing many of the most commonly used abbreviations and internet expressions many still remained. These could have played a significant part in BERT's understanding of the data. If a method was developed to also replace these, it would be interesting to compare those results with those achieved here. One thing that has become conclusive is the fact that replacing instances of abbreviations and slang does not automatically make the model more efficient in detecting cyber grooming.

5.3 Balancing the data set

One of the major problems with the data set collected from AIBA AS is that it is quite imbalanced. The ratio of of predatory to innocent conversations is 5,8%. While this ratio is much lower in the complete PAN12 data set, the number of conversations are far greater, so the BERT model is trained to recognize a far larger number of predatory conversations. This large imbalance could be an explanation to why the model seemed to be overfitting the data during training. To help mitigate this, new data could used during this training process, but may be difficult to achieve. Due to the sensitive nature of cyber grooming it is difficult to aquire new data sets that could be used to train machine learning models in detecting predatory conversations. Apart from the PAN12 data set, I have not been able to come across any other relevant data sources that others have used in their research. To my knowledge, the data that has been made available by AIBA As is the only similar other data set, how ever it is not publicly available.

Due to this scarcity, it would be helpful to see if techniques such as oversampling or synthetically generating new predatory conversations to help balance out the data sets, in hopes that it might lead to an improvement in performance in BERTs ability to detect cyber grooming.

5.3.1 Oversampling the data

The first and most simple solution to this problem would be directly oversampling the data set. This would mean selecting out random predatory conversations from the already existing training set and directly duplicating them. This would lead to the same conversations appearing more than once. The results from this process can be seen here.

Epoch	Train Loss	Val Loss	Precision	Recall	F1
1	0.3565	0.2261	0.0000	0.0000	0.0000
2	0.1946	0.1852	0.0000	0.0000	0.0000
3	0.1549	0.1744	0.0000	0.0000	0.0000
4	0.1154	0.1736	0.8000	0.1429	0.2424
5	0.0725	0.1636	0.8000	0.2857	0.4211
6	0.0319	0.1855	0.7647	0.4643	0.5778
7	0.0180	0.2008	0.7059	0.4286	0.5333
8	0.0072	0.2273	0.6842	0.4643	0.5532
9	0.0039	0.2332	0.7059	0.4286	0.5333
10	0.0039	0.2357	0.7500	0.4286	0.5455

Table 5.8: Training results with tripled amount of predatory data

Epoch	Train Loss	Val Loss	Precision	Recall	F1
1	0.3333	0.2152	0.0000	0.0000	0.0000
2	0.1994	0.1894	0.0000	0.0000	0.0000
3	0.1594	0.2273	0.0000	0.0000	0.0000
4	0.1387	0.1629	0.0000	0.0000	0.0000
5	0.0764	0.1904	0.8889	0.2857	0.4324
6	0.0516	0.2185	0.8333	0.1786	0.2941
7	0.0212	0.2166	0.8182	0.3214	0.4615
8	0.0065	0.2402	0.6000	0.4286	0.5000
9	0.0037	0.2547	0.7857	0.3929	0.5238
10	0.0037	0.2534	0.7500	0.4286	0.5455

Table 5.9: Training results with four times the amount of predatory data

When comparing these results from the ones derived from training the model with the original number of conversations several things can be said. The overall F1-score of the models where not improved and the validations losses still increased at a similar rate to the initial model. This may indicate that a straight forward oversampling method will not improve BERT' chances of detecting cyber grooming and it will not mitigate overfitting of the data.

5.3.2 Back translating data

Another technique that could be used to generate new predatory conversations is back translation. This is a technique used in natural language processing to help balance data sets, thus hopefuly improving the performance of a maachine learning model. This is a process where text is translated into a another language and then translated back . This generates a slightly different text, but it keeps roughly the same meaning.

This was implemented using the Hugging Face transformers library. The code used for back translation utilized two pre-trained models from the library - one for English to French translation and another for French to English translation. These models are 'Helsinki-NLP/opus-mt-en-fr' and 'Helsinki-NLP/opus-mt-fr-en' respectively. The library provides functionalities to load these models along with their corresponding tokenizers. These models are then used to perform the translation tasks in the back translation augmentation. How this was implemented through code can be seen in the appendix.

The results from the BERT models performance when using back trainslations can be seen in this table.

Epoch	Train Loss	Val Loss	Precision	Recall	F1 Score
1/10	0.2774	0.1927	1.0000	0.2143	0.3529
2/10	0.1409	0.1757	1.0000	0.2500	0.4000
3/10	0.0968	0.1489	0.8750	0.2500	0.3889
4/10	0.0615	0.1682	0.9000	0.3214	0.4737
5/10	0.0351	0.2060	0.8000	0.4286	0.5581
6/10	0.0166	0.1888	0.8235	0.5000	0.6222
7/10	0.0114	0.1866	0.7368	0.5000	0.5957
8/10	0.0070	0.2158	0.8000	0.4286	0.5581
9/10	0.0043	0.2197	0.6818	0.5357	0.6000
10/10	0.0042	0.2204	0.6818	0.5357	0.6000

Table 5.10: Training results with back translation augmentation

The results achieved from back translating the predatory conversations are quite different compared to the other oversampling methods. They achieve a higher overall F1-score than just simply oversampling the same conversations. During the first two epochs the model had a perfect precision, meaning that every conversation that it flagged as predatory was indeed predatory. Unfortunately it was not able to detect the majority of predatory conversations in the data set as evident in the low recall score.

5.4 Conclusion of performance improvement

During this chapter several methods were explored as to how one could improve upon BERT base's ability to detect cyber grooming, both on a general basis and in the context of abbreviation use.

Three different models where tested to see how their performance would stack up against the first BERt base model. The conclusion there was that BERT base outperformed all three of these models. With RoBERTa coming the closest in it's ability to detect cyber grooming based on F1-scores.

When it came to the usage of abbreviations in chats, BERT showed that it was able to detect cyber grooming at a higher rate in conversations where the original messages were kept the same. When many abbreviations and other forms of slang where replaced, this did not automatically result in a better performance for BERT. This indicated that BERT has a good understanding of the English language, even when it appears in a more informal way. To get a more fair comparison how ever, it would be interesting to see the performance of BERT with a higher rate of replacement for the abbreviations, slang and other obfuscations in the conversations. This is due to the fact that many of these were still relatively prevalent in the messages even after the replacements where done.

Efforts to balance out the data set where also taken. This is due to the fact that the predatory conversations where highly outnumbered in the data sets. When the BERT model was trained on the original predatory messages and the ones that were back translated, it was able to improve it's precision rates drastically compared to the previous models only trained on the original number of predatory conversations.

Chapter 6
Discussion

When analyzing the results from the research conducted up against the research questions several things can be said.

The first of these questions was:

1. How may BERT be trained in order to better understand abbreviations and the meaning or context behind them in ongoing chats?

In order to answer this question a initial model was trained as a base line. This model was trained on using a data set from AIBA AS. No changes were made in the data when training this first model. This models ability to detect cyber grooming was compared to all the other models trained.

When trying to train a model to better detect cyber grooming with an emphasis on abbreviation use, changes in the training and test data where made. In both of them, a large amount of abbreviations and other forms of internet slang were replaced with more formal language. A BERT model was then trained on this new data. The performance of the model, indicated that BERT did not perform better when the informal language was replaced. When this did not yield any positive results, only replacing the abbreviations that were either present or not present in BERT's tokenizer were tried. Both of these experiments did not show an improvement on BERT's inital performance.

When trying to balance the number of predatory messages to innocent ones, back translation was used. This synthetically generated new predatory conversations by translating them into French and back again to English. When training the BERT model on this new data, a remarkably high precision score was achieved compared to the first BERT model. During the first epochs of training it scored a precision scores between 90 and 100%.

When trying to fine tune BERT to better understand emojis in the context of detecting cyber grooming, the remaining research questions where answered. These being:

1. How important is the usage of abbreviations for BERT when it is trying to detect cyber grooming? Do any of these more strongly imply that a predatory conversation is or is not taking place?
2. Does replacing an abbreviation with it's original form change how BERT analyzes the chat?

It would appear from the results when trying to improve the performance of the models, that abbreviations do not pose a significant challenge to BERT when it is trying to detect cyber grooming. This also goes for the last research question. When replacing the abbreviation and slang instances with their expansions or formal meanings. it did not improve BERT's performance. On the contrary, it seemed to decrease its abilty to detect cyber grooming. This fact can also be seen in the performance of BERT when detecting cyber grooming when trained on the similarly sized data sets from AIBA AS and PAN12. Despite the AIBA AS conversations containing a larger amount of abbreviaions and slang, the models achieved roughly the same results.

This could be due to a several factors. One of these could be due to the fact that BERT already has a good understanding of the English language, it is able to leverage this knowledge when training and develop an understanding of more informal forms of the language. Another reason it is seemingly able to understand abbreviations is due to its bidirectionality. When reading and trying to understand a sentence, it does an analysis from both right to left and left to right. This gives it a deeper understanding of language compared to other natural language processing models that only read the text from left to right[Tou19]. This could led to BERT understanding the meaning of the sentence even though it does not understand the abbreviation. By looking at the other words in the sentence, and the sentences before and after, it could be able to derive a good enough understanding of what is happening in the ongoing conversation.

On the other hand, not all instances of abbreviations and internet slang were replaced. Through the remaining examples present in the conversations, it is possible that BERT has able to train itself to better understand informal English. Due to the many different variations of the text in the conversations is proved difficult to develop a method that would replace all of these instances of informal English. If that had been done, BERT would in theory have no way of training itself on abbreviaitons and slang. This would could then give a better idea of how much of a role abbreviations and slang plays a part in BERT's ability to detect cyber grooming.

When answering the question if any abbreviations could more strongly than others if a predatory conversations is taking place or not, some trends appear. In both the PAN12 and AIBA AS data sets the abbreviaiton GF seemed to be overrepresented in predatory converstions. It appeared far more frequently in these conversations, despite innocent conversations vastly outnumbering predatory ones in both data sets. This could be expected as conversations discussing subjects that involve themes such as relationships and love could be more likely to contain some form of grooming. An interesting discovery here is that the same cannot be said for the abbreviation BF, meaning boyfriend. In both the PAN12 and AIBA AS data sets, it was more commonly used in innocent conversations.

Another issue that was encountered during this research was that the validation results all seemed to increase after training the BERT models over several epochs. This meant that overfitting was taking place. This means that the model is only learning to recognize the training set, thus not being able to generalize that well. This could mean that it would not be able to perform well when trying to detect cyber grooming on new data. Some measures where taken in order to try and mitigate this such as adjusting parameters such as sequence length and batch sizes during training, but they did not provide any noteworthy changes.

Chapter 7 Conclusion

7.1 Future research

Based on some of the remarks made in the discussion chapter I believe that I have found some potential topics for future research that may build upon what has been discovered during this thesis.

A problem during this research is that I was not able to discover a quick and efficient way to translate all of the informal language present in the AIBA AS chats into more formal English. This meant that even though I had replaced a large amount of abbreviaions and slang with their formal meaning, their was still a large amount present in the conversations used in training and testing. In order to fully understand how much of a role such language plays a role in BERT's ability to detect cyber grooming it would be helpful if a method was developed that could easily translate it before was processed by BERT. This would make it easier to do a comparison of BERTs performance on the different forms of language.

Another topic that could be interesting to explore further is trying to add weight or extra significance to words and expressions in the conversations being analyzed by BERT. My work mostly focused on developing a model, changing a few parameters and seeing how the model performed on several similar data sets. Tailoring BERT to be more sensitive to certain words, expressions, phrases or abbreviations could have a significant effect on BERTs performance when trying to detect cyber grooming. The list of abbreviations and table of the number of times they occurred in both innocent and predatory conversations could be a relevant starting point when conducting such research. Another abbreviation that was more common in predatory messages in the AIBA AS data set was rp, meaning role play. This also could be expected as the act of role play does have some sexual connotations. I think it is worth mentioning that this analysis is somewhat speculative. It does not capture all of the abbreviations and slang words present in the PAN12 and AIBA AS data set. It would be far more helpful to conduct such an analysis with a larger amount and variety of abbreviations and expressions, especially with those that could have a more sexual or predatory meaning. Most of abbreviations and slang that have been tested for in this research have had a fairly general meaning and could be naturally used in a variety of conversations.

7.2 Summary of findings

To summarize the findings of this thesis, research was done on the natural language processing model BERT and it's ability to detect cyber grooming, with an emphasis on the usage of abbreviations and internet slang. The result of this was that BERT was able to detect cyber grooming at a higher rate when the data included more instances of informal language. I found this to be suprising as I thought that the data that BERT was originally trained on was far more formal than the language present in the cyber grooming data sets.

This research may have some implications in a broader context as well. It has shown that the presence of informal language does not seem to have a major affect on BERT's ability to detect cyber grooming. This means that it has to have a relatively good understanding of the english language. This goes to show the potential that BERT has as a natural language processing model. Without any changes made to the architecture, a few lines of code and a relatively small data set, it arguably gained a solid grasp and understanding of a new sub genre of the English language, this being internet slang. This power could potentially be leveraged for other purposes as well. During this research I believe that BERT has demonstrated that it has the potential to be a powerful tool when it comes to analyzing chats. Instead of cyber grooming, I believe that it could also be used for analysis and detection of other phenomena in online chats.

As for the research questions that were set out be answered at the beginning of this thesis, I feel that they have been answered somewhat sufficiently. This being said, I struggled to fine tune a BERT model to better detect cyber grooming specifically based on the presence of abbreviations and slang. No architectural changes where made to the existing BERT base model, and no major parameters were changed in order to achieve this goal. The only thing that was changed substantially to achieve this was the data being used. A significant improvement in the precision of the model was achived when using back translation to help balance out the data set.

While this gave insight into how BERT was able to detect cyber grooming based in the general presence of abbreviations or slang, I feel that I was not able to make any significant changes to the BERT model in order to make it process the abbreviations and slang differently than the other language. If I was to fine tune the model during this experiments, I felt that I would be doing it on a general basis, and not in the context of the usage of abbreviations and slang.

That being said, I have discovered that BERT is able to detect cyber grooming despite the text including a large amount of informal language. Removing abbreviations and slang with more formal words will not automatically improve upon BERT's ability to detect cyber grooming. This shows that BERT has a solid understanding of human language, as it is able to understand the naunces and meaning of conversations even though it appears in a more informal form. This fact can be used for future analysis of texts similar to the conversations used during this research.

Chapter A Appendix

A.1 Code used to train model

Listing A.1: Python code used to train BERT base

```
import torch
from torch.utils.data import Dataset, DataLoader
from transformers import BertTokenizer, BertForSequenceClassification
import pandas as pd
import numpy as np
from sklearn.metrics import accuracy_score, precision_score,
    recall_score, f1_score,roc_auc_score, average_precision_score
from transformers import AdamW, get_linear_schedule_with_warmup

MAX_LEN = 128 #Maximum length of input tokens
BATCH_SIZE = 16 #Batch size for training and evaluation
TOKENIZER_NAME = 'bert-base-uncased'
path = "/path/to/data" # replace with your data path
save_directory = "/path/to/save_directory" # replace with your save
    directory
tokenizer = BertTokenizer.from_pretrained(TOKENIZER_NAME)

class ChatDataset(Dataset):
    def __init__(self, filename, tokenizer, max_length):
        self.tokenizer = tokenizer  # BERT tokenizer
        self.data = pd.read_csv(filename)  # Load data from CSV
        self.text = self.data.message  # Extract chat messages
        self.labels = self.data.label  # Extract labels
        self.max_length = max_length  # Maximum token length

     # This method returns the total number of samples in the dataset
    def __len__(self):
        return len(self.text)

    # This method formats a single sample for model input
    def __getitem__(self, idx):
        text = str(self.text[idx])
        label = self.labels[idx]
        # Encode the text into tokens, attention masks, etc.
        encoding = self.tokenizer.encode_plus(
            text,
            add_special_tokens=True,
            max_length=self.max_length,
            return_token_type_ids=False,
            padding='max_length',
            return_attention_mask=True,
            return_tensors='pt',
            truncation=True
        )

        # Return the encoding and the label
        return {
            'input_ids': encoding['input_ids'].flatten(),
            'attention_mask': encoding['attention_mask'].flatten(),
            'label': torch.tensor(label, dtype=torch.long)
        }

# Initialize the tokenizer for BERT
TOKENIZER_NAME = 'bert-base-uncased'
tokenizer = BertTokenizer.from_pretrained(TOKENIZER_NAME)

# Instantiate the dataset and dataloader
train_dataset = ChatDataset(path+"/training.csv", tokenizer, max_length
    =MAX_LEN)
test_dataset = ChatDataset(path+"/testing.csv", tokenizer, max_length=
    MAX_LEN)

train_dataloader = DataLoader(train_dataset, batch_size=BATCH_SIZE,
    shuffle=True)
test_dataloader = DataLoader(test_dataset, batch_size=BATCH_SIZE,
    shuffle=False)

model = BertForSequenceClassification.from_pretrained('bert-base-
    uncased', num_labels=2)
device = torch.device('cuda' if torch.cuda.is_available() else 'cpu')
model.to(device)

epochs = 10 #Number of training epochs
lr = 2e-5 #Learning rate
num_training_steps = epochs * len(train_dataloader) #Total number of
    training steps
num_warmup_steps = int(0.1 * num_training_steps)  # Using a 10% warmup

optimizer = AdamW(model.parameters(), lr=lr)
scheduler = get_linear_schedule_with_warmup(optimizer, num_warmup_steps
    , num_training_steps)

# Train the model
def train(model, dataloader, optimizer, scheduler, device):
    model.train() #Set model to training mode
    total_loss = 0 #Initialize total loss

    for batch in dataloader:
        input_ids = batch['input_ids'].to(device)
        attention_mask = batch['attention_mask'].to(device)
        labels = batch['label'].to(device)

        model.zero_grad() # Reset gradients
        outputs = model(input_ids, attention_mask=attention_mask,
            labels=labels)
        loss = outputs.loss
        loss.backward()

        total_loss += loss.item()
        optimizer.step() #Update weights
        scheduler.step() #Update learning rate

    return total_loss / len(dataloader) #return average loss

def evaluate(model, dataloader, device):
    model.eval() # Set model to evaluation mode
    total_loss = 0

    for batch in dataloader:
        input_ids = batch['input_ids'].to(device)
        attention_mask = batch['attention_mask'].to(device)
        labels = batch['label'].to(device)

        with torch.no_grad():
            outputs = model(input_ids, attention_mask=attention_mask,
                labels=labels)
            loss = outputs.loss

        total_loss += loss.item()

    return total_loss / len(dataloader) #return average loss

# Create a directory to save the model and tokenizer
save_directory = DIRECTORY TO BE SAVED

# Save the model and tokenizer
model.save_pretrained(save_directory)
tokenizer.save_pretrained(save_directory)

# Define a function to get predictions from the model
def get_predictions(model, dataloader, device):
    model.eval() #Set model to evaluation mode
    predictions = []
    true_labels = []

    for batch in dataloader:
        input_ids = batch['input_ids'].to(device)
        attention_mask = batch['attention_mask'].to(device)
        labels = batch['label'].to(device)

        with torch.no_grad():
            outputs = model(input_ids, attention_mask=attention_mask)
            logits = outputs.logits

        preds = torch.argmax(logits, dim=1).cpu().numpy()
        labels = labels.cpu().numpy()

        predictions.extend(preds)
        true_labels.extend(labels)

    return np.array(predictions), np.array(true_labels)


best_f1 = 0

for epoch in range(epochs):
    train_loss = train(model, train_dataloader, optimizer, scheduler,
        device)
    val_loss = evaluate(model, test_dataloader, device)

    # Get predictions and true labels for the validation set
    predictions, true_labels = get_predictions(model, test_dataloader,
        device)

    # Calculate precision, recall, and F1-score
    precision = precision_score(true_labels, predictions)
    recall = recall_score(true_labels, predictions)
    f1 = f1_score(true_labels, predictions)

    # If the F1 score has improved, save the model and tokenizer
    if f1 > best_f1:
        best_f1 = f1
        model.save_pretrained(save_directory)
        tokenizer.save_pretrained(save_directory)

    print(f"Epoch:␣{epoch+1}/{epochs},␣Train␣Loss:␣{train_loss:.4f},␣
        Val␣Loss:␣{val_loss:.4f},␣Precision:␣{precision:.4f},␣Recall:␣{
        recall:.4f},␣F1:␣{f1:.4f}")
```

A.2 Abbreviations replaced in the conversations

Here is a overview of all of the abbreviations and slang phrases with their expansions or formal meanings.

Table A.1: Abbreviations and their expansions

Abbreviation	**Meaning**
wby	what about you
hbu	how about you
rp	role play
idc	I do not care
ik	iknow
yk	you know
hru	how are you
btw	by the way
wyd	what are you doing
u	you
ur	your
gm	good night
rn	right now
brb	be right back
idk	I do not know
alr	all right
af	as fuck
ash	as hell
tbh	to be honest
wanna	want to
r	are
fr	for real
lol	laughing out loud
sm	so much
np	no problem
ty	thank you
ofc	of course
sc	Snapchat
plz	please
nah	no
auto	autograph

Continued on next page

Table A.1 – continued from previous page

Abbreviation	**Meaning**
afk	away from keyboard
wtf	what the fuck
pics	pictures
pic	picture
tysm	thank you so much
b4	before
rlly	really
irl	in real life
jk	just kidding
y	why
pm	personal message
lemme	let me
ngl	not going to lie
thx	thanks
smth	something
lgtm	looks good to me
nvm	never mind
gf	girlfriend
bf	boyfriend
x	kiss
xx	kisses
ml	my love
bby	baby
disc	discord
wth	what the hell
inv	invite
txt	text
gimme	give me
wym	what do you mean
tho	though
lmao	laughing my ass off

Continued on next page

Table A.1 – continued from previous page

Abbreviation	Meaning
smh	shaking my head
ight	all right
xo	hugs and kisses
x o	hugs and kisses
x-o	hugs and kisses
m8	mate
ic	I see
ily	I love you
ppl	people
yolo	you only live once
#	number
cs	because
ttyl	talk to you later
sup	Whats up
yh	yeah

A.3 Number of abbreviation replacements in data

Here is an overview of how many times each individual abbreviaiton or slang phrase appeared in the PAN12 and AIBA AS data set

A.3.1 Abbreviaiton count in PAN12 data set

Table A.2: Abbreviation count in the PAN12 data set

Abbreviation	Innocent Counts	Predatory Counts	Total Counts
wby	1	0	1
hbu	46	0	46
rp	1	0	1
idc	2	0	2
ik	5	0	5
yk	0	0	0
hru	3	0	3
btw	29	0	29

Continued on next page

Table A.2 – continued from previous page

Abbreviation	**Innocent Counts**	**Predatory Counts**	**Total Counts**
wyd	0	0	0
u	2828	501	3329
ur	387	37	424
gm	0	0	0
rn	1	0	1
brb	27	2	29
idk	24	1	25
alr	0	0	0
af	3	0	3
ash	0	0	0
tbh	1	0	1
wanna	209	9	218
r	519	33	552
fr	4	0	4
lol	355	17	372
sm	0	0	0
np	47	0	47
ty	21	0	21
ofc	3	0	3
sc	2	0	2
plz	101	0	101
nah	27	0	27
auto	7	0	7
afk	5	0	5
wtf	35	0	35
pics	85	1	86
pic	44	0	44
tysm	0	0	0
b4	6	3	9
rlly	0	0	0
irl	4	0	4

Continued on next page

Table A.2 – continued from previous page

Abbreviation	**Innocent Counts**	**Predatory Counts**	**Total Counts**
jk	11	0	11
y	45	2	47
pm	6	2	8
lemme	3	1	4
ngl	0	0	0
thx	66	1	67
smth	0	0	0
lgtm	0	0	0
nvm	11	0	11
gf	11	28	39
bf	6	0	6
x	53	0	53
xx	13	0	13
ml	1	0	1
bby	1	0	1
disc	0	0	0
wth	4	0	4
inv	0	0	0
txt	2	0	2
gimme	3	0	3
wym	0	0	0
tho	14	0	14
lmao	19	0	19
smh	0	0	0
ight	4	0	4
xo	2	0	2
x o	0	0	0
x-o	0	0	0
m8	1	0	1
ic	3	1	4
ily	3	0	3

Continued on next page

Table A.2 – continued from previous page

Abbreviation	Innocent Counts	Predatory Counts	Total Counts
ppl	8	0	8
yolo	0	0	0
#	11	0	11
cs	0	0	0
ttyl	27	4	31
sup	44	3	47
yh	2	0	2

A.3.2 Abbreviaiton count in AIBA AS data set

Table A.3: Abbreviation count in the AIBA AS data set

Abbreviation	Innocent Counts	Predatory Counts	Total Counts
yolo	0	0	0
auto	103	4	107
hru	138	17	155
gm	4	1	5
fr	265	5	270
lol	1557	65	1622
hbu	119	31	150
sc	126	22	148
b4	8	0	8
ash	10	1	11
wyd	163	17	180
gf	66	5	71
x	83	14	97
alr	93	10	103
ml	4	1	5
yk	134	33	167
brb	39	8	47
x-o	10	0	10
cs	7	3	10
sup	19	3	22

Continued on next page

Table A.3 – continued from previous page

Abbreviation	**Innocent Counts**	**Predatory Counts**	**Total Counts**
disc	39	0	39
wby	32	5	37
inv	63	5	68
yh	31	6	37
af	45	8	53
lgtm	0	0	0
afk	19	1	20
idc	32	17	49
pm	10	1	11
wym	57	3	60
r	301	70	371
txt	7	2	9
nah	197	22	219
ngl	94	3	97
plz	33	7	40
smth	11	1	12
tbh	110	7	117
xo	3	0	3
nvm	87	14	101
ty	185	13	198
ight	24	3	27
jk	36	1	37
ofc	129	16	145
tho	324	28	352
gimme	14	4	18
pic	25	8	33
m8	0	0	0
smh	43	9	52
tysm	72	5	77
btw	89	13	102
rp	25	27	52

Continued on next page

Table A.3 – continued from previous page

Abbreviation	**Innocent Counts**	**Predatory Counts**	**Total Counts**
x o	0	0	0
lmao	350	36	386
xx	22	0	22
ik	182	15	197
wth	17	3	20
rn	305	22	327
idk	461	39	500
irl	47	7	54
sm	28	4	32
ur	1062	272	1334
wtf	37	9	46
ttyl	11	2	13
y	202	22	224
bf	68	19	87
ppl	87	13	100
#	17	4	21
ic	5	0	5
pics	8	2	10
lemme	47	8	55
wanna	375	92	467
ily	43	1	44
np	74	10	84
thx	65	6	71
bby	15	2	17
rlly	66	11	77
u	4317	941	5258

References

[18] «Do perverted justice chat logs contain examples of overt persuasion and sexual extortion?: A research note responding to chiang and grant 2017 and 2018.», English, *Language and Law/Linguagem e Direito*, vol. 5, no. 1, pp. 97–102, Jul. 2018.

[BK19] P. Bours and H. Kulsrud, «Detection of cyber grooming in online conversation», in *2019 IEEE International Workshop on Information Forensics and Security (WIFS)*, 2019, pp. 1–6.

[Edu20] I. C. Education, *Natural Language Processing (NLP)*, https://www.ibm.com/cloud/learn/natural-language-processing, [Online; accessed 06-November-2022], 2020.

[Eli19] K. Ó. Elisabeth Staksrud, «Tilgang, bruk, risiko og muligheter. norske barn på internett. resultater fra eu kids online-undersøkelsen i norge 2018. eu kids online og institutt for medier og kommunikasjon, universitetet i oslo», 2019.

[GKS12] A. Gupta, P. Kumaraguru, and A. Sureka, «Characterizing pedophile conversations on the internet using online grooming», *CoRR*, vol. abs/1208.4324, 2012. [Online]. Available: http://arxiv.org/abs/1208.4324.

[IC12] G. Inches and F. Crestani, «Overview of the International Sexual Predator Identification Competition at PAN-2012», in *CLEF 2012 Evaluation Labs and Workshop – Working Notes Papers, 17-20 September, Rome, Italy*, P. Forner, J. Karlgren, and C. Womser-Hacker, Eds., CEUR-WS.org, Sep. 2012. [Online]. Available: http://www.clef-initiative.eu/publication/working-notes.

[LCG+20] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, *Albert: A lite bert for self-supervised learning of language representations*, 2020.

[LH19] I. Loshchilov and F. Hutter, «Decoupled weight decay regularization», in *International Conference on Learning Representations*, 2019. [Online]. Available: https://openreview.net/forum?id=Bkg6RiCqY7.

[LOG+19] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, *Roberta: A robustly optimized bert pretraining approach*, 2019.

[Sco19] W. Scott, *TF-IDF from scratch in python on a real-world dataset.* https://towardsdatascience.com/tf-idf-for-document-ranking-from-scratch-in-python-on-real-world-dataset-796d339a4089, [Online; accessed 06-November-2022], 2019.

[SDCW20] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, *Distilbert, a distilled version of bert: Smaller, faster, cheaper and lighter*, 2020.

[SQXH19] C. Sun, X. Qiu, Y. Xu, and X. Huang, «How to fine-tune bert for text classification?», in *Chinese Computational Linguistics*, M. Sun, X. Huang, H. Ji, Z. Liu, and Y. Liu, Eds., Cham: Springer International Publishing, 2019, pp. 194–206.

[Tou19] J. D. M.-W. C. K. L. K. Toutanova, «Bert: Pre-training of deep bidirectional transformers for language understanding», 2019.

[TOYS20] T. Tomihira, A. Otsuka, A. Yamashita, and T. Satoh, «Multilingual emoji prediction using bert for sentiment analysis», *Int. J. Web Inf. Syst.*, vol. 16, pp. 265–280, 2020.

[VLA21] M. Vogt, U. Leser, and A. Akbik, «Early detection of sexual predators in chats», Jan. 2021, pp. 4985–4999.

[VSP+17] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, «Attention is all you need», 2017.

[YKC20] J. Yadav, D. Kumar, and D. Chauhan, «Cyberbullying detection using pre-trained bert model», in *2020 International Conference on Electronics and Sustainable Communication Systems (ICESC)*, 2020, pp. 1096–1100.

[Zho19] V. Zhou, *A Simple Explanation of the Bag-of-Words Model*, https://victorzhou.com/blog/bag-of-words/, [Online; accessed 06-November-2022], 2019.