Can ChatGPT Generate Code to Support a System Sciences Bachelor's Thesis?

Kan ChatGPT generera kod för att stödja en kandidatuppsats i systemvetenskap?

Authors: Solin Amin and Johan Hellström

Supervisor: Arend Hintze

Examinator: Joonas Pääkkönen

Thesis for Bachelor Degree in Informatics; 15 Credits; First Cycle [SWE: Examensarbete för filosofie kandidatexamen i Informatik, 15hp]

2023-06-10

Published in full text, freely available

Abstract

Background — ChatGPT is a chatbot released in November 2022. Its usage has grown to include being used in academia and for scientific writing, with varying results. We investigate if ChatGPT can be used for the technical part in a Bachelor's thesis in System Sciences.

Aim — We evaluate if it is possible to generate the code for detecting potential gender bias in previous responses from ChatGPT, in the form of a dialogue.

Method — We use an exploratory case study where an iterative dialogue with ChatGPT is used to generate Python code to be able to analyse previous responses made by ChatGPT. The methods for development were chosen by the authors from suggestions by ChatGPT.

Results — Two separate dialogues resulted in a program that combined a fine-tuned Natural Language Processing model together with sentiment analysis and word frequency analysis. The program successfully identified responses in the dataset as having a female or male gender bias or being gender neutral.

Conclusions — ChatGPT serves as a powerful tool for coding, although it currently falls short of being a one-stop solution that can generate code sufficient for more complex tasks with a single prompt. Our experience suggests that ChatGPT accelerates one's work when the user possesses some programming knowledge. With further development, ChatGPT could transform coding workflows and increase productivity in related fields.

Implications — ChatGPT as a tool is very capable in supporting students in the technical aspect of a Bachelor's thesis and it is not unreasonable to assume that it works in other contexts, as well. As such, one can achieve more with the tool than without, and consequently it would be for the better to integrate ChatGPT into thesis work. This stresses the point that we need to find better regulations for cheating and plagiarism.

Keywords — AI, ChatGPT, NLP, Python

Abstract

Bakgrund — ChatGPT är en chatbot som släpptes den 22 november 2022. Sedan dess har dess användningsområden växt till att inkludera den akademiska världen och vetenskapligt skrivande, med varierande resultat. Vi undersöker om ChatGPT kan användas för den tekniska delen av en kandidatexamen i systemvetenskap.

Syfte — Vi utvärderar om det är möjligt att i en dialogform generera kod för att upptäcka potentiell könsbias i tidigare svar från ChatGPT.

Metod — Vi använder en utforskande fallstudie där en iterativ dialog med ChatGPT används för att generera Python-kod för att kunna analysera tidigare svar från ChatGPT. Utvecklingsmetoderna valdes av författarna utifrån förslag från ChatGPT.

Resultat — Två separata dialoger med ChatGPT resulterade i ett program som kombinerade en finjusterad Natural Language Processing-modell med stämnings- och ordfrekvensanalys. Programmet identifierade svar i datasetet med att ha kvinnlig eller manlig könsbias, eller att vara könsneutralt.

Slutsatser — ChatGPT är ett kraftfullt verktyg som kan användas för programmering. I dagsläget är ChatGPT ingen komplett lösning som kan generera kod tillräcklig för mer komplexa uppgifter med en enda prompt. Vår erfarenhet visar att ChatGPT accelererar ens arbete då användaren besitter viss kunskap inom programmering. Vid fortsatt utveckling kan ChatGPT ombilda programmeringsflöden och öka produktiviteten i relaterade områden.

Följder — ChatGPT som verktyg är mer än kapabelt med att stödja studenter med den tekniska delen av ett examensarbete, det är heller inte orealistiskt att anta att det är möjligt att även använda det i andra sammanhang. Med detta sagt kan man utföra mer med verktyget än utan, och följaktligen är det till det bättre att integrera ChatGPT i examensarbeten. Detta driver på poängen att vi behöver finna en lösning vad gäller reglering och hantering av plagiat.

Nyckelord — AI, ChatGPT, NLP, Python

Table of contents

Abbreviations and definitions

Term	Definition
AI	Artificial intelligence, a field of computer science focused on creating machines that can perform tasks that typically require human intelligence.
AI4EU	An AI on-demand platform and ecosystem, funded by the European Union.
API	Application Programming Interface, enables different software to communicate with each other.
Auto-GPT	AI agent that uses the OpenAI API to automatically break down a task into subtasks. It can create and revise prompts to manage new information gathered from ChatGPT.
BERT	Bidirectional Encoder Representations from Transformers, a language model developed by Google AI Language for understanding context in text.
Chatbot	Computer program designed to simulate conversation with human users.
ChatGPT	Generative Pre-trained Transformer (GPT), and AI language model for interactive conversations. It generates human-like text responses using NLP techniques.
DistilBERT	A smaller and faster version of the BERT language model.

Gender bias	Prejudice or discrimination based on gender, often leading to unequal treatment or opportunities.
GPT	Generative Pre-trained Transformer, a type of language model developed by OpenAI. It utilises machine learning to generate human-like text, having been trained on a diverse range of internet text. The model can perform tasks such as answering questions, writing essays, and summarising texts. Since the release of GPT-1 in 2018, there have been several improved iterations: GPT-2 in 2019, GPT-3 in 2020, GPT-3.5 in 2022 and GPT-4 in March 2023.
Language model	Statistical models that predict the probability of a word or sentence given the preceding context.
LLaMA	Large Language Model Meta AI, a language model developed by Meta AI released in February 2023.
ML	Machine Learning, a subfield of AI that involves building algorithms and models that can learn from and make predictions on data, without being explicitly programmed.
NLP	Natural Language Processing, the field of AI focused on human language understanding.
NLTK	Natural Language Toolkit, a Python library for NLP.
Pandas	A Python library for data manipulation and analysis.

Prompt	The instruction given to ChatGPT from the user.
Parameter	An internal variable that a language model or any ML model has learned during the training process.
RoBERTa	Robustly Optimized BERT Pretraining Approach, an optimised version of the BERT language model.
Sentiment analysis	Analysis of text to determine if the emotion of the text is positive, negative, or neutral.
Supervised fine-tuning	The process of using labelled datasets to generate target tasks to improve learning.
Unsupervised pre-training	The process of training models on texts from a large corpus to learn general language patterns.
Word frequency analysis	Analysis of the number of times a certain word, or words, appear in a piece of text.

1 Background

1.1 Introduction to ChatGPT

On November 30, 2022, OpenAI released their Large Language Model (LLM), ChatGPT, to the public. ChatGPT is a chatbot that can answer general questions about a wide range of subjects. It is also able to generate code in various programming languages. It is most proficient in the Python programming language (OpenAI, n.d.-b).

Interacting with ChatGPT is done on a website as a prompt, where the user enters its input into a text field and the response from ChatGPT appears underneath it. Depending on the question given by the user, the answers, or responses, given by ChatGPT are long and informative. The model can also create new text with fact-filled content from the users' requirements or improve on an already written text. ChatGPT has the ability to write different types of code, including code for data collection from various sources, code for data analysis as well as visualisation of the collected data.

The knowledge of current events is something that ChatGPT 3.5 does not have, since it is limited to information that it was trained with, which is up until September 2021 (OpenAI, n.d.-c).

ChatGPT's abilities come from the GPT-3.5 and GPT-4 models, which are generative pre-trained transformers (GPT). The GPT models are trained with a combination of unsupervised pre-training and supervised fine-tuning. The regular version of ChatGPT is fine-tuned from GPT-3.5 (OpenAI, 2023).

The unsupervised pre-training was done on long unlabelled contiguous texts from a large corpus to allow the generative model to learn on extended information (Radford et al., 2018). It also uses a transformer decoder, a variant of the transformer, during the pre-training. The transformer architecture was introduced in 2017 by Vaswani et al. and is using a mechanism called self-attention, which allows the model to weigh the importance of different words in a given context which makes it effective in understanding[1] and generating natural language. After the pre-training, the supervised fine-tuning uses labelled datasets to be able to generate the target tasks to improve the learning. (Radford et al., 2018).

GPT-1 in 2018 introduced the idea to use a semi-supervised approach for training the language model. Previously most deep learning methods required large amounts of labelled data, which lessens their applicability in domains where such data do not exist, whereas GPT-1 used a combination of a large corpus of unlabelled text and several datasets with manually annotated training examples (Radford et al., 2018). Development continued and in 2019 GPT-2 was introduced. While most prior language models were trained on data such as newspaper articles, Wikipedia, or fiction books, the approach to GPT-2 was to have a vastly diverse dataset.

[1] The model does not understand per se, but rather predicts the words to follow, which the user perceives as it understands the context.

This included books, as well as web scrapes, where they focused on content that was curated or filtered by humans. Compared to GPT-1, which had 117 million parameters, GPT-2 had an increase of its parameters to 1.5 billion (Radford et al., 2019). This greatly improves the model's performance.

The GPT-3 model introduced in 2020 was a further improvement from the GPT-2 model and was trained on an even larger set of data. When evaluated on a dataset called "TriviaQA", which is used for reading comprehension and question answering, the GPT-3 model had an accuracy of over 70%, which means that the GPT-3 model could quite accurately generalise answers to the trivia questions contained in the dataset without being trained on it. By increasing the size of the model, specifically the number of parameters, its performance significantly improved, and GPT-3 had an increase of its parameters to 175 billion (Brown et al., 2020).

Other examples of Large Language Models are BERT, developed by Google AI Language with 340 million parameters (Devlin et al., 2018) and LLaMA (Large Language Model Meta AI) developed by Meta with 65 billion parameters (Touvron et al., 2023).

The difference in accessibility between the earlier GPT models and ChatGPT is significant. GPT-3, for instance, was intended to be used by developers that accessed the model through OpenAI's API. Even though other Large Language Models existed when ChatGPT was released, the difference besides its design, training data and fine-tuning, was that ChatGPT was released globally and people all over the world could easily interact with it through OpenAI's website.

The public interest for this new AI chatbot was high and in less than a week it already had one million users (Altman, 2022). The media were quick to pick up the interest of this new phenomenon. Technology magazine Wired writes that ChatGPT has "become the darling of the internet since its release last week" (Knight, 2022). They further write how users are "enthusiastically" posting their experiences with ChatGPT to write essays, answering complex coding problems and even creating literary parodies, such as writing a biblical verse in the style of the King James Bible about how to remove a peanut butter sandwich from a VCR (Ptacek, 2022). Despite the many abilities of ChatGPT, it is far from perfect and if the knowledge about a subject is missing from the model, it is prone to "fabricate convincing-looking nonsense on a given subject" (Knight, 2022).

In an article from December 5, 2022, The New York Times calls it "the best artificial intelligence chatbot ever released to the general public" (Roose, 2022). They further write that it is "ominously good" at answering open-ended questions, often found on school assignments. They also state in the article that educators are predicting that tools like ChatGPT are the end for homework and take-home exams (Roose, 2022).

Even as impressive as ChatGPT is and how popular it became in such a short time, the technology is not without concerns. In early January 2023, public schools in New York City restricted access to ChatGPT on its devices and networks, citing concerns for the negative impact ChatGPT

might have on the students' learning abilities and concerns about the validity of the content generated by ChatGPT (Wiggers, 2023). As some schools also implement restrictions to the access of ChatGPT with reasonings around academic dishonesty, others disagree.

In another New York Times article with the title "Don't Ban ChatGPT in Schools. Teach With It." (Roose, 2023), the author acknowledges the ethics problem surrounding texts generated by ChatGPT and whether the information given is correct. However, instead of a ban he suggests that the model can be used as a teaching aid and be treated as how schools treat a calculator: allowed for some assignments, but not for others. As a teaching aid, the author notes that ChatGPT can be used for outlining essays where the students then finish the essay longhand. One teacher who had tried this method said the process deepened the students' understanding of the topic and also taught them how to interact with an AI model to get a helpful response (Roose, 2023).

Anders Enström, a teacher in Huddinge, Sweden, uses ChatGPT in his classroom and in an interview with the trade journal *Skolvärlden*, he voices his opinion that he believes that it will fundamentally change the school. He is concerned that criticism of the schools' digitisation could lead to a ban but believes that the new tools will give students access to explanations and help them shift from consumers to producers. Enström acknowledges the risk of cheating with the chatbot but suggests that teachers can manage the risk by assigning other types of tasks or locking students' devices to certain pages. He believes that the old educational ideal of lecturing people and telling them what to think is being replaced and that students should develop the ability to empathise with others (Olsson, 2023).

ChatGPT is used for scientific writing, as well. Entire articles have been authored with the help of ChatGPT. For instance, an article with the title "ChatGPT and the Future of Medical Writing", published in the medical journal Radiology was partly written by ChatGPT (Biswas, 2023). ChatGPT was given headings and subheadings as prompts and then the contents generated by ChatGPT were edited by the human author. In Russia, as reported by The Moscow Times, a student used ChatGPT to write a thesis in 23 hours (The Moscow Times, 2023), and in a Twitter thread (Zhadan, 2023) the student gives a detailed outline of how he proceeded with the work.

It is evident that ChatGPT is a powerful tool for certain tasks, but when it comes to its reliability to generate truthful information it is still lacking. An article titled "ChatGPT in Scientific Writing: A Cautionary Tale" (Zheng & Zhan, 2023), published in The American Journal of Medicine used ChatGPT to try to evaluate its ability to generate information about an article it did not have any knowledge about (because of its information cut-off in September, 2021).

The authors of the "ChatGPT in Scientific Writing: A Cautionary Tale" article summarised key facts from the article and repeatedly prompted ChatGPT with one question to evaluate its responses. What they discovered was that the answers from ChatGPT were well written and sounded plausible, but the generated responses contained information that was

plain wrong and also contained information that was made up by ChatGPT (Zheng & Zhan, 2023). The authors note that the "falsifications" and "fabrications" are not easily noticeable for readers or inexperienced reviewers. They further state that, since the way ChatGPT generates its responses, they are constructed as its own "story", without consideration for logic or accuracy of this said story, and that in its current form it is not ready for scientific writing. They further state that the scientific writers that rely on ChatGPT must manually verify the information and references generated by ChatGPT and because of this there is no "obvious advantage to writing with ChatGPT". The authors further discuss the ethics surrounding scientific research, that poor data management and fabricated findings are considered serious ethical violations in scientific research and that the responsibility for the accuracy and integrity lies on the listed authors of an article, and not ChatGPT.

The article "ChatGPT or academic scientist? Distinguishing authorship with over 99% accuracy using off-the-shelf machine learning tools" (Desaire et al., 2023) discusses the potential of AI-generated writing and its impact on academic and professional writing. It focuses on differentiating AI-generated text from human-generated text in academic writing, which was not a significant threat before the release of ChatGPT. The article presents a method to discriminate between text generated by ChatGPT and human scientists, with claims of an accuracy rate of over 99%. The method uses 20 different features, including paragraph length and the use of equivocal language. This targeted approach could be useful in detecting AI usage in academic writing and other fields. The article also highlights the limitations of previous studies and suggests a need for a careful reassessment of the best way to distinguish advanced language models like ChatGPT from human writing (Desaire et al., 2023).

An analysis of the creative performance of GPT language models on tests measuring convergent (each question has one "correct" answer) and divergent (multiple correct answers may exist) creativity from 2019 to 2023 was conducted and published online with the title "Exploring Creativity in Large Language Models: From GPT-2 to GPT-4" (Jun, 2023). The test comprised of connecting three unrelated words with a fourth one, generating alternative uses for everyday objects, and listing ten different nouns that differ as much as possible from each other. The analysis revealed that there were cases where the GPT-3.5 and GPT-4 models matched or even surpassed human creativity scores. The conclusion of the article is that accurately measuring creativity using text-based tests is intricate and alternative testing methods must be developed. Additionally, the article highlights that GPT models' performance on these tests may be influenced by their exposure to the task during training (Jun, 2023).

1.2 How does ChatGPT work?

To generate a response ChatGPT first takes the input text from the user and breaks it down to smaller units, such as words, characters or subwords, and converts them to numerical representations called tokens (Sennrich et al., 2015).

The input tokens are processed to capture their context within the input sequence, and the model predicts the most probable next token based on patterns learned during its training (Radford et al., 2018).

To be able to generate human-like responses, ChatGPT uses something called self-attention from the transformer architecture (Vaswani et al., 2017). This attention function can be described as mapping a query and a set of key-value pairs to an output, where the query, keys, values, and output are all vectors. The attention function is applied in parallel across several layers and multiple attention heads, which allows the model to capture different aspects of the context and relationships between tokens in the input sequence.

After the attention function stage, ChatGPT generates a sequence of tokens based on a probability distribution created by the model. During this step, the tokens are generated by the model one at a time until a predefined number of tokens, or an end-of-sequence token, is reached (Radford et al., 2018).

The last step is de-tokenization where the model converts the sequence of tokens back into human-readable text. The de-tokenization involves reversing the tokenization process by mapping the generated tokens back to their corresponding words, characters, or subwords, which then are presented to the user (Sennrich et al., 2015).

Figure 1 illustrates the process of generating an answer from ChatGPT given the input from the user.

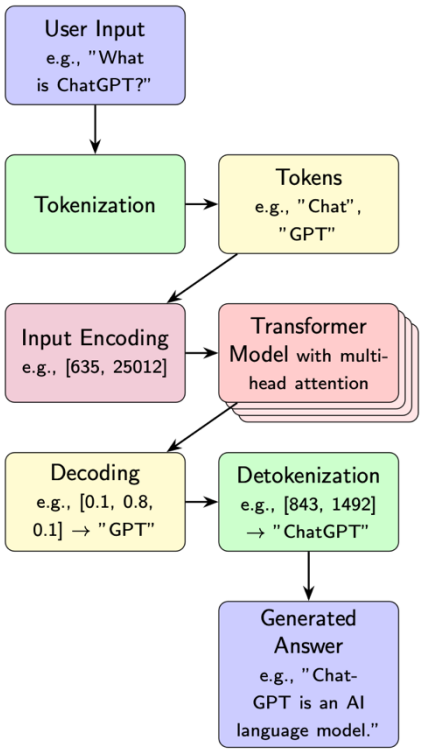

Figure 1: ChatGPT response generation.

1.3 Problem formulation

The increasing sophistication of text generation models, such as ChatGPT, has sparked concerns about their potential misuse and the need to understand their limitations. In this thesis, we evaluate the capabilities of ChatGPT in exploring if it can write the coding part of a Bachelor's thesis in System Sciences. To achieve this, we generate and evaluate code using ChatGPT 3.5 and the output is evaluated based on the time spent and its performance. The study uses an exploratory approach, and the topic of gender bias serves as a test case to evaluate ChatGPT 3.5's ability to generate code.

As for the specific topic of gender bias, it is important to note that this could have been any other topic. The original question would remain the same: Can ChatGPT Generate Code to Support a System Sciences Bachelor's Thesis? Thus, "gender bias" only serves as an example, and we are not concerned with the outcome of that analysis. We are only concerned whether or not the code works from a technical point of view, and if results can in principle be obtained.

Gender bias as explained in the article "Gender Bias in Text: Origin, Taxonomy, and Implications" (Doughman et al., 2021) explores the pervasive issue of gender bias in AI applications, specifically within NLP and ML systems. The detrimental consequences of gender bias are examined, including its role in perpetuating inequality, widening gender wage gaps, and restricting the representation of women in leadership positions. The article highlights the concerning ability of AI systems to reflect and amplify gender biases and stereotypes inherited from historical training data. To address this issue, the authors propose a comprehensive taxonomy that categorises gender bias into various types, such as generic pronouns, sexism, occupational bias, exclusionary bias, and semantics (Doughman et al., 2021).

It is important to note that the topic of gender bias is only a generic topic used to test ChatGPT 3.5's ability to generate code for a specific purpose, and we are not seeking to mitigate any biases. Through the evaluation of the generated code, we aim to gain insights into the potential benefits and limitations of using ChatGPT in generating code for academic purposes.

1.3.1 Expectations

In an ideal scenario the functionality of the code to be generated by ChatGPT would be perfect, requiring no further intervention from the authors. This might not be the case, and that it is necessary to have some back-and-forth dialogue with ChatGPT. Furthermore, we might also have the case where we never arrive at a functioning result and, the dialogue might never end. Also, that the resulting code might be of such low quality that it would have been more efficient to have written it manually without any input from ChatGPT.

If someone without any experience from programming attempted this process, it might result in an endless dialogue because of this lack of programming experience. However, having experience in programming in

Python or any other programming language can help steer the dialogue to not end up in this never-ending dialogue.

1.4 Scope and limitations

It appears that the usage of ChatGPT is increasing in many fields. This is a complicated situation with many questions regarding consequences that have not yet been answered. As students we would like to investigate what this means for us in our specific situation, which is writing a Bachelor's thesis.

The purpose of this thesis is to assess if it is possible to use ChatGPT in a dialogue form to conduct the coding part of a Bachelor's thesis in System Sciences. The purpose is to assess in our specific case, the possibility of producing code to detect gender bias in responses from ChatGPT.

This study focuses on generating and evaluating code for a quantitative computational study on gender bias in a large language model, as a part of a Bachelor's thesis in System Sciences, using ChatGPT 3.5. The study does not focus on mitigating any biases or conducting any experiments related to gender bias. The results of this study provides insights into the potential benefits and limitations of using large language models in generating code for academic purposes.

Given time and resource constraints, this study does not explore other models or algorithms beyond ChatGPT. Furthermore, only a limited amount of generated text will be investigated, which means that the results cannot be generalised to all types of text generation that can be done by ChatGPT.

It should be noted that this study does not investigate the use of ChatGPT or any other language model for academic dishonesty or cheating purposes. While this is an important topic, it falls outside the scope of our research question and objectives. Additionally, we will not be examining any academic policies or interventions related to the use of language models in academic settings. Our focus is solely on the use of ChatGPT as a tool for assisting students in the process of writing the code for a Bachelor's thesis. Finally, the study focuses on the English language and Python as the programming language, which means that the results may not be generalisable to other natural languages or programming languages.

By investigating the effectiveness of an iterative dialogue with ChatGPT, we contribute to the understanding of the potential in how this language model can be used in generating code for a Bachelor's thesis.

ChatGPT is only used for content generation in the form of Python code. We do not use ChatGPT in any other form to help us write this thesis.

2 Method

2.1 Literature review

Since ChatGPT is a new phenomenon, there do not exist many peer-reviewed references to be used as a basis for our work. The literature we gathered is mostly based on opinion pieces describing ChatGPT from different perspectives and technical articles describing the different technologies behind the ChatGPT language model.

As ChatGPT was released in November 2022, we have used search engines such as DuckDuckGo and Google together with date restrictions to get search results in a chronological order to discover and follow the reception and different opinions surrounding ChatGPT in a more structured way. Since many of our sources are non-peer-reviewed opinion pieces, we have limited ourselves to more well-renowned sources for these opinions. We have also critically reviewed them to obtain as fair a view of the topic as possible. For our peer-reviewed sources we used Google Scholar and Summon.

2.2 Research strategy

This thesis is an exploratory case study to understand ChatGPT in-depth as a tool and how it can affect different areas, such as academia in our case. Our primary objective is to determine if ChatGPT can generate code to detect gender bias in a dataset containing previous responses made by ChatGPT. We accomplish this by interacting with ChatGPT in a dialogue form to generate Python code to analyse a given dataset.

The interaction with ChatGPT is ground-breaking in the sense that users conduct a dialogue with something not human, but still get human-like responses. Currently there exist no protocols or procedures for this, and an exploratory case study approach allows for an in-depth examination of the phenomenon (Oates, 2006). The dialogue in its entirety can be found online at Figshare (Hellström, 2023).

2.2.1 Data collection

The primary dataset consists of the dialogue we engage in with ChatGPT, including both our queries and the system's responses. The dialogue is time-stamped to be able to assess the time spent generating the functional code with ChatGPT. This method of data generation does not fit the traditional methods such as interviews or observational methods, as it involves engaging with ChatGPT instead of human participants. This process is not always straightforward since the responses from ChatGPT varies depending on how the given prompt is formulated, which further cements the exploratory process involved in this thesis.

Our secondary data come from repositories on Huggingface and GitHub with datasets of collected responses from ChatGPT. These were found by using search engines with key words related to datasets and ChatGPT.

The dataset that we analyse contains previous responses that are generated by ChatGPT and covers a broad number of topics. Below follows two example texts from the dataset.

> 1: "Square diamonds are commonly referred to as princess cut diamonds. The princess cut is a popular choice for diamond engagement rings and other jewelry because it offers a modern and sophisticated look. It is a square-shaped diamond with pointed corners and a unique, stepped facet pattern. The princess cut is known for its excellent sparkle and brightness, making it a popular choice for those who appreciate a more sparkling and lively diamond."

> 2: "The Mona Lisa is a famous painting that was created by the artist Leonardo da Vinci. It is considered a masterpiece of art because it is a very detailed and realistic painting that is also full of mystery and emotion. Many people find the Mona Lisa to be a beautiful work of art and are fascinated by the story behind it. The Mona Lisa is also one of the most famous paintings in the world, which makes it special and valuable."

2.2.2 Generating code

To obtain the code for analysing the previous responses from ChatGPT, we engaged in a structured dialogue with ChatGPT, where the dialogue was designed to provide clear instructions with specified requirements. This back-and-forth prompting with ChatGPT allowed for iterative refinement of the generated code to suit our objective. The code that ChatGPT generates through the dialogue allows us to fine-tune an NLP model which we combine with sentiment analysis and text frequency analysis.

In Figure 2, we see an example of how this dialogue took place.

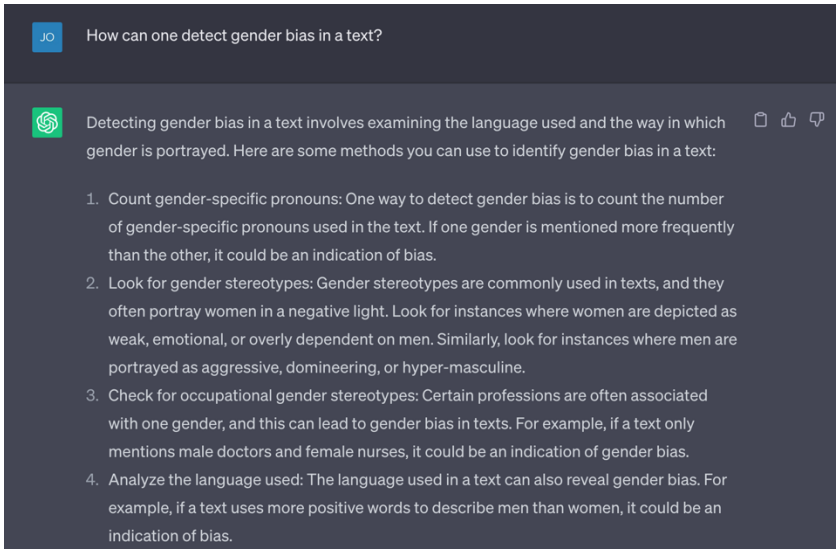

Figure 2: First interaction with ChatGPT.

2.2.3 Evaluating and correcting code

The generated code was executed against a dataset named "ChatGPT-RetrievalQA" (Askari et al., 2023) that contain responses from ChatGPT. This enabled us to identify issues with the code, such as syntax errors, logical flaws, and other inaccuracies. On the occasions where the generated code was non-functional, we documented the specific issues and again prompted ChatGPT to revise the code. In cases where we could not reach a solution to an error with ChatGPT, we manually reviewed the code to be able to find what was causing the error, and consequently fixed it. All problems were eventually solved.

Figure 3 illustrates an example of a response from ChatGPT containing Python code.

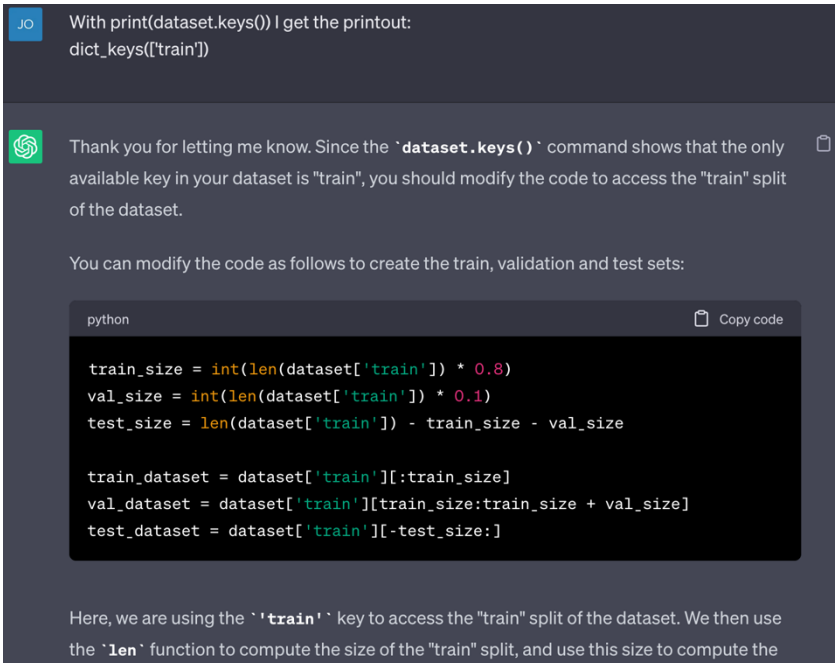

Figure 3: Response from ChatGPT containing code.

3 Results

3.1 Preface

The results in this section come from a highly dynamic and exploratory method. The work done was not possible to standardise, since we could not know what the questions would be beforehand. For instance, we could not formulate a questionnaire and then see how the results would turn out.

The following results represent our written record of our interaction with ChatGPT and the dialogue in its entirety can be found at Figshare (Hellström, 2023).

3.2 Training an NLP model

We started the dialogue with ChatGPT by prompting it on how one could detect gender bias in a piece of text. As a response we got a list of different methods, ranging from counting gender-specific words to analysing the specific words used in the text. It also emphasised that detecting gender bias was not a straightforward process and could require careful analysis and interpretation.

In the second prompt we asked if it would be possible to use NLP tools to detect potential gender bias, and the response was another list of suggestions, where we followed up with another prompt asking for an NLP model to do the detection. This is shown in Figure 4.

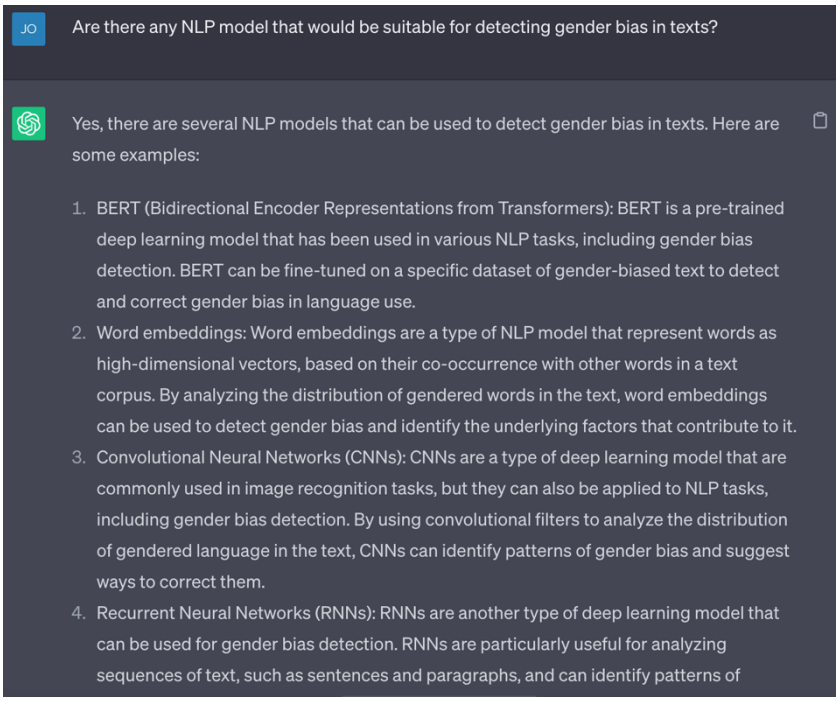

Figure 4: NLP suggestion from ChatGPT.

The response ChatGPT gave us from this prompt included a suggestion to use a model called BERT (Bidirectional Encoder Representations from Transformers) with the explanation: "BERT is a pre-trained deep learning model that has been used in various NLP tasks, including gender bias detection" and that it could be fine-tuned on "dataset of gender-biased text to detect and correct gender bias in language use".

An NLP model can be used to interpret human language so that a computer might understand it, not only the meaning of the words and phrases used, but also the sentiment and the intent behind them. The NLP model BERT is used by Google Search for improving the way one can interact with it (Nayak, 2019).

We prompted ChatGPT again with a request to use a dataset from Huggingface called "md_gender_bias[2]" to train an NLP model. The model we chose is an optimised version of the BERT model called RoBERTa (Liu et al., 2019). The response from ChatGPT this time was a long string of Python code to implement the training of the model on the dataset. We took this code response and added it into a Jupyter Notebook where we would execute the code. This version of the code did not work, and we started the back-and-forth process with ChatGPT to try to generate the first functional piece of code. After some investigation of the dataset, we realised that we had to specify a subset of the dataset, since otherwise it would default to a subset that did not suit our specific needs. This process from zero knowledge in how to train a model and for the functional code to run took roughly 4 hours during the span of 2 days.

What we discovered during this process was that every response from ChatGPT was very confident. Even if the generated code did not work as intended, the response was constructed as if it works. This, in combination with our lack of experience with this type of programming, sometimes made the process cumbersome.

Since the use of RoBERTa demanded us to fine-tune a model, which is computationally heavy, we decided to try another approach to detect gender bias before committing to the idea of training the model. The second approach was sentiment analysis coupled with word frequency analysis which was suggested by ChatGPT. The development of this piece of code went well and we decided to train the RoBERTa model to then try to combine the two methods.

[2] "The Multi-Dimensional Gender Bias Classification dataset is based on a general framework that decomposes gender bias in text along several pragmatic and semantic dimensions: bias from the gender of the person being spoken about, bias from the gender of the person being spoken to, and bias from the gender of the speaker." (*Md_gender_bias · Datasets at Hugging Face*, n.d.)

3.3 Word frequency analysis and sentiment analysis

In the second dialogue we prompted ChatGPT with the question in how one could detect gender bias in an unlabelled dataset containing rows of text. The response from this prompt differed from the previous one where we only mentioned gender bias in texts.

The suggestions this time included word frequency analysis (how many times a word, or words, occur in a text), gender association analysis, sentiment analysis (whether a sentence is either positive or negative), and human annotation. We continued to prompt ChatGPT with question regarding how to approach the topic and after a while of back-and-forth interactions we reached a conclusion to use word frequency analysis in combination with sentiment analysis to detect potential gender bias in the unlabelled dataset.

With the prompt where we stated that we wanted to use the library NLTK (Natural Language Toolkit) for sentiment analysis and the Pandas library for word frequency analysis, the response from ChatGPT was a functional Python program that implemented the techniques we requested and from this response we started the back-and-forth process of trying to adjust the code to suit our needs for the thesis work.

3.4 Combining an NLP model with sentiment analysis and word frequency analysis

The next step was to implement the model that we had fine-tuned on the "md_gender_bias" dataset. We prompted ChatGPT about how to combine the sentiment analysis and word frequency analysis code together with the pre-trained model. This is shown in Figure 5.

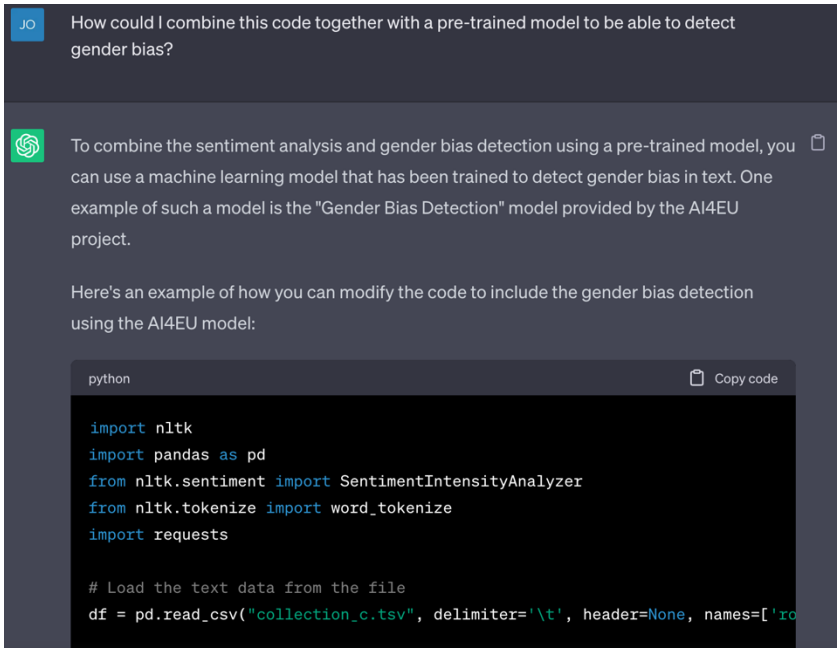

Figure 5: Combining an NLP model with our code.

The response from ChatGPT was a suggestion to use a gender bias detection model from "AI4EU", where it generated code to access an API to

analyse the text. This code looked as it would work, but when executed we got an error message that the host was not known. Either the API was not in use any longer, or it might not have existed in the first place. As described by Zheng and Zhan (2023), the responses from ChatGPT are constructed as its own "story", without consideration for logic or accuracy. We cannot confidently determine whether this is the case with the missing API.

Since we already had saved our fine-tuned NLP model that we wanted to use we did not pursue any troubleshooting regarding this, and we prompted ChatGPT with this information. The first response contained code that would enable us to import the fine-tuned model for use with the program. With this first code snippet we got an error message related to it not being compatible with our model. After a few prompts we had sorted out the error and ChatGPT correctly suggested a method to use to be able to import the fine-tuned model.

After importing the fine-tuned model, a longer session of debugging together with ChatGPT started, since we got many different errors with the code. One problem that took a considerable amount of time to debug was when a mismatch in tensor sizes sometimes occurred during calculations related to the sentiments of the input texts.

The mismatch error led us to try to figure out another way to calculate the sentiment scores, and after some back-and-forth dialogue with ChatGPT and our own search for a different solution, we settled on a pre-trained model called DistilBERT (Sanh et al., 2019). After prompting ChatGPT that we wanted to use this specific model it generated a code snippet that at first seemed to work without problems, but when running it on a larger set of samples from our dataset the mismatch error kept occurring.

The dialogue with ChatGPT continued with the back-and-forth theme to solve the mismatch error with different methods, with examples of truncating the input text to only analyse a certain number of characters from it, or using "sliding window" techniques, where the input text would be handled in smaller chunks to calculate the sentiment values. The error persisted and ChatGPT continued to suggest solutions which often led to changing larger pieces of the code without improvement. The problem was resolved through manual debugging and not with the assistance of ChatGPT. The root cause of the error was that the code did not account for two special tokens that are added to the input sequence by the tokenizer encoder, one at the beginning and one at the end. These tokens are used by the fine-tuned model during processing. The RoBERTa model has a maximum input length of 512 tokens, including these special tokens. The code failed to account for these special tokens and caused the input to occasionally exceed this limit, leading to the error.

Once we had the fully functional code, we were able to analyse our entire dataset with previous responses made by ChatGPT. We noticed, however, that the scores related to gender bias were very similar, and even responses without any use of gendered words were labelled as containing a gender bias. To create an alternative positive control, we prompted

ChatGPT to write three paragraphs in the style of a thrilling novel: one gender-neutral, one with a strong female bias and one with a strong male bias. These sentences were not used for training the model, and after reading them we intuitively agreed that ChatGPT indeed had generated one gender-neutral, one with a female bias, and one with a male bias. The RoBERTa model trained on the Huggingface dataset identified those three paragraphs as containing a gender bias with a high bias score, even the gender-neutral sentence, which clearly indicated that there were some issues with the fine-tuned model.

We prompted ChatGPT about the issue with the high bias scores, and the response from ChatGPT contained suggestions that there might be bias in the training data, ambiguity in the sentences analysed, or limitations in the model used for detecting gender bias. The issue persisted, and we discussed it with our supervisor during a meeting where we were suggested that it might be a hyperparameter problem. Hyperparameters are values chosen before the training of the model, such as learning rate or number of epochs (each time the training dataset is passed through the learning algorithm), and they control the learning behaviour of the model in question. We re-trained the model with different hyperparameters but still got the erroneous results when analysing our dataset.

The cause of the problem with the gender bias score was identified by revisiting the dataset used to fine-tune the RoBERTa model, we discovered that there were three labels present (gender-neutral, female, and male) and the code generated by ChatGPT assumed that there were only two (female and male). We manually modified the training code for the model to handle the correct labels and started the re-training to get a model that would perform correctly.

With the new model we had to adjust the code to be able to identify potential gender bias, since we now had three labels used for classification of the texts. Again, we started the process of back-and-forth code adjustment with ChatGPT to get an end-result of a functional program that appeared to correctly identify whether a text was gender biased or not, based on the criteria defined in the program. The code generated by ChatGPT was well structured and easy to read.

3.5 Accuracy and time estimate

In Table 1 we have the estimates of the time spent developing the functional code together with ChatGPT, the total number of code responses that we executed, and the amount of them that executed without any errors.

Table 1: Time estimates and code generation.

Model training		
Total number of prompts to ChatGPT	Number of code responses	Functioning code responses
86	50	7
Estimate of time spent in total	4 hours 57 minutes	
Sentiment & word frequency analysis		
Total number of prompts to ChatGPT	Number of code responses	Functioning code responses
126	43	23
Estimate of time spent in total	11 hours 21 minutes	

When developing the code for fine-tuning the NLP model, we spent just below 5 hours with a total of 86 prompts to ChatGPT. Of these responses 50 contained code, and 7 of those code responses executed without any issues. For the development of the sentiment analysis and word frequency analysis program that we combined with the fine-tuned NLP model, we spent just above 11 hours. This was the result of a total of 126 prompts to ChatGPT. Of the responses 43, contained code and 23 of them contained functional code that we refined by the back-and-forth dialogue to get the code to function.

Table 2 shows results from evaluating the model after correcting the labels used when fine-tuning. The table contains the original fine-tuned model and the one trained with the correct labels.

Table 2: Fine-tuned model evaluation.

Model	Accuracy	F1	Precision	Recall
Original	0.1456	0.0370	0.0212	0.1456
Corrected	0.8924	0.8766	0.8696	0.8924

As a measure in how well the code ChatGPT generated performed, we evaluated the models with the test set from the training dataset. When looking at the score of the metric *accuracy* and comparing the two models, we see that it greatly differs between the original and the corrected model. During evaluation, the original fine-tuned model made the correct predictions 14% of the time compared to 89% for our corrected one. This suggests that the generated code was functional and demonstrated the predicted behaviour during the process of fine-tuning the model.

Figure 5 shows the learning process of the model during the training process and how its ability to correctly classify data changes with the evaluation step.

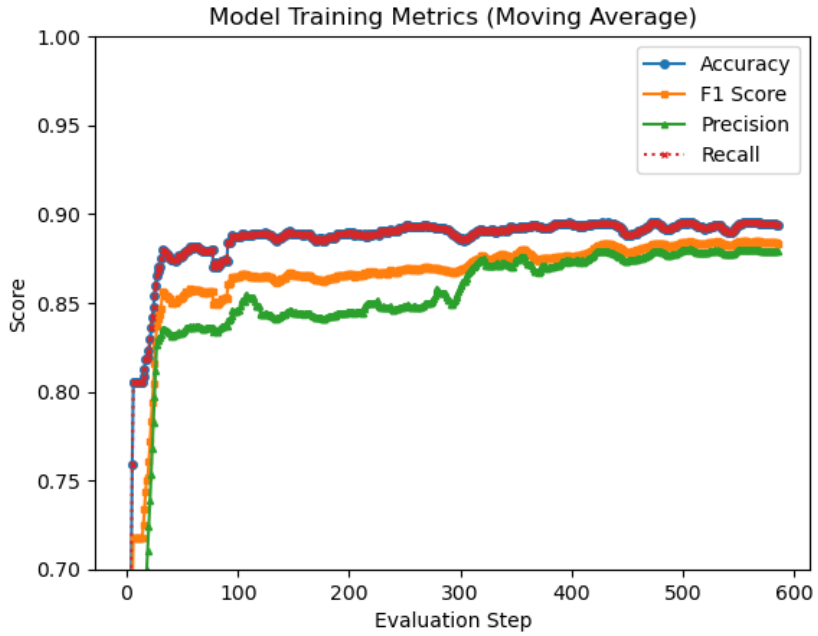

Figure 5: Model Training Metrics.

When the completed program was executed on our dataset with previous responses from ChatGPT, we obtained a total of 26,883 results. 25,172 of them were identified as "gender-neutral", where the categorised text did not contain any of the gender-specific words defined by ChatGPT in the program. The other two labels "female" and "male" resulted in 239 identified texts with a female gender bias and 1,472 with a male gender bias. Given gender bias detection's complexity, including both overt gendered language and subtler stereotypes (Doughman et al., 2021), we cannot ensure that the ChatGPT-generated program captures all bias types.

4 Discussion and Conclusions

4.1 Findings

For our thesis work we were interested in evaluating the possibility of using ChatGPT as a tool to develop code for a Bachelor's thesis in System Sciences. In our case we used an example of detecting potential gender bias in a dataset with previous responses made by ChatGPT. The method used for doing this was of an exploratory nature where we interacted with ChatGPT in the form of two back-and-forth dialogues.

We started off the first dialogue by asking questions related to gender bias, and how one would be able to detect said gender bias. With the suggestions from ChatGPT we decided to try to implement a combination of two methods for gender bias detection, using a fine-tuned Natural Language Processing model together with sentiment analysis and word frequency analysis. To structure the development in a clearer way, we found it appropriate to use two separate dialogues: one for the development to fine-tune an NLP model, and one for the sentiment analysis and word frequency analysis.

What we found out quite early on was that ChatGPT tended to generate large pieces of code all at once, which sometimes made it hard to quickly grasp what had changed between the responses. When executing the code responses from ChatGPT they were not always functioning, but by using the back-and-forth style of dialogue we gradually transformed the code to a fully working program that implemented the fine-tuned NLP model combined with sentiment analysis and word frequency analysis.

In one of the dialogues, we developed a program to fine-tune an NLP model with a dataset suited for the thesis. Despite having no previous experience in this, and with the help of ChatGPT, we were able to start the training of the model in just under 5 hours, which we estimate is not very long given the complexity of the topic.

The second dialogue took another path to the problem of detecting gender bias in our dataset. From the suggestions of ChatGPT we started out by developing a program that used sentiment analysis and word frequency analysis. When we had a functioning program doing this, we started the implementation of our fine-tuned model. This second part of the development took a considerable amount of time, where on one occasion ChatGPT was not able to identify the cause of a coding error and we had to manually debug the code to make it work. Another issue during the development was an oversight from us where we had assumed that one part of the code generated by ChatGPT was functioning as expected although it was not. There was an issue with the labels in the dataset used for fine-tuning the model. The code generated by ChatGPT only took two of the three existing labels into consideration, which caused the model to perform poorly. Once the cause was discovered and the code adjusted, we saw a large improvement in its performance after re-training the model.

To put everything into perspective, it is important to consider the time commitment that this project would have required without the assistance of ChatGPT. This includes reading and understanding all necessary documentation, writing, and debugging all the code, and interacting with our supervisor. To make an educated guess of the time spent, it is likely that we would not have achieved a finished result in less than 40 hours of work.

If we had relied entirely on ChatGPT to generate and debug our code, the 20 hours spent could have increased significantly. In our case we noticed when ChatGPT got stuck and could not progress. If this was a limitation of ChatGPT itself or if it was caused from the way our prompts were formulated, we do not know. Currently it seems as it is not yet possible to complete a more complex project without manual intervention during the process. We believe that by engaging in dialogues with ChatGPT, we have gained a better understanding of this topic.

To effectively code with ChatGPT, having some knowledge about programming, is indeed beneficial. While you do not need to be a programmer, a basic understanding of programming concepts is recommended. The minimum required level of programming know-how to utilize ChatGPT would involve familiarity with fundamental programming concepts such as variables, loops, conditionals, and functions. Additionally, understanding how to work with strings and handle user input is advantageous.

Those with no experience in programming might find it challenging to write a working program interacting with ChatGPT, since they lack the basic knowledge about programming concepts. Asking ChatGPT to write a simpler program would probably be possible even for someone that does not have any programming experience, but it is unclear if it would save the person time compared to manually searching for information.

Our conclusion is that ChatGPT is an advanced tool that can be iteratively used to reach an end-goal. Our process involved a back-and-forth dialogue, which we believe could potentially be universally applied and generalised to suit other fields of study, although we did not verify this in our study. Currently, ChatGPT is not a one-stop solution that can generate code sufficient for more complex tasks with a single prompt, as we experienced in our work. However, with our existing programming knowledge, and through our iterative dialogues, we were able to create properly functioning code. Our experience suggests that ChatGPT accelerates a user's workflow given some programming knowledge, and presumably, this efficiency increases with greater programming expertise.

4.2 Limitations and scope of research

Our chosen method was of an exploratory nature which is not possible to generalise beforehand. This affects our results in the way that the interactions with ChatGPT generates different responses depending on how the

user prompt is formulated. With a deeper knowledge about how to interact with ChatGPT, we believe that the work done in this Bachelor's thesis could have been conducted in an even more efficient way.

A notable limitation to consider is ChatGPT's knowledge cut-off, established in September 2021. The field of software development, in our case related to machine learning, is an ever advancing one. In our thesis work the code was developed without the knowledge about the progression of the field since September 2021. This not to say that our result would have necessarily been better or different. Nevertheless, we still find it worth mentioning that ChatGPT 3.5 does not have access to the most current information because of this cut-off date from September 2021, which in some cases might be a limitation.

After our dialogue with ChatGPT was complete, we have been made aware of a tool named "Auto-GPT". This is an AI agent that uses the OpenAI API with either the GPT-4 or GPT-3.5 models, to automatically break down a task into subtasks, where it can create and revise prompts to manage new information (Auto-GPT, n.d.). We cannot speculate on how the use of a tool like this would have affected our work, but it shows that the ways in which people can interact with ChatGPT or the GPT models are quickly evolving.

In March 2023 OpenAI introduced plugins for ChatGPT to enhance its capabilities. The plugins allow ChatGPT to access developer defined APIs to perform different actions such as searching the internet, booking flights, accessing real-time stock market information, and knowledge-based information from companies, such as internal documents (OpenAI, n.d.-a). Currently, the function of adding plugins to ChatGPT is in a beta stage and not available for all users. Additionally, this feature is limited to the GPT-4 version of ChatGPT. With the addition of plugins that can access information ChatGPT was not trained on, the model's capabilities will be significantly enhanced, enabling it to provide more accurate, up-to-date, contextually relevant responses.

In our thesis work we had the aim to evaluate if ChatGPT would be able to generate code for the technical part of a Bachelor's thesis. Through the dialogues with ChatGPT we reached the result of a functioning program to detect potential gender bias in a dataset with previous responses from ChatGPT. We cannot say with certainty whether or not the program correctly identifies gender bias. Nevertheless, this falls outside the scope of this thesis. Detecting gender bias is a complex process, and our program can at most be seen as a base for further development to reach this goal.

4.3 Further research

4.3.1 Programming with ChatGPT

There are two dimensions to take into consideration when using a back-and-forth dialogue with ChatGPT: user skill and the complexity of the problem. Someone with no previous programming experience will most probably become stuck and will not be able to solve the problem in question. Nevertheless, someone with a high level of programming skill might

immediately find the issues with the code and correct it themselves. It seems obvious that with simpler problems ChatGPT might give more complete answers, and with more difficult problems the dialogue might be much longer to reach a result. These two dimensions of user skill and task complexity are something that affected our results but were not something we investigated.

To investigate the correlation between user skill and task complexity, and how these factors influence the ease or difficulty of using ChatGPT for solving coding tasks, one could engage students with varying levels of programming skills. These students would then solve programming tasks of different complexities using ChatGPT. By doing so, we could gain more definitive answers to questions regarding the time and effort required to solve programming tasks with the assistance of ChatGPT.

4.3.2 Gender bias

While our thesis primarily focused on the coding aspect of a Bachelor's thesis in System Sciences, specifically for a quantitative computational study on gender bias in large language models, it is crucial to acknowledge the significance of the gender bias topic and the need for further investigation. Our decision to prioritise the coding part was driven by the constraints of time and scope for this particular study. However, it is essential for future research to delve deeper into the gender bias issue and explore its implications in greater detail.

Gender bias in AI, particularly in LLMs like ChatGPT 3.5, has gained considerable attention in recent years. The potential for AI systems to reflect and amplify biases inherited from historical training data is a critical concern that can perpetuate inequality, reinforce stereotypes, and hinder progress towards gender equality. By dedicating more resources and effort to researching gender bias, we can contribute to a more comprehensive understanding of the problem and develop effective strategies for addressing it.

Future research on gender bias should consider various aspects, such as the underlying causes and origins of bias in training data, the impact of biased AI systems on societal perceptions and behaviours, and the development of fair and unbiased algorithms and models. This could involve exploring different data pre-processing techniques to mitigate bias during model training, incorporating diverse and inclusive datasets, and investigating the social and cultural factors that contribute to bias in language models.

By expanding the scope and depth of research on gender bias in large language models, we can contribute to the advancement of AI technologies that are more equitable, inclusive, and aligned with the principles of social justice. Future researchers have an opportunity to make significant strides in this field and pave the way for a fairer and more unbiased AI ecosystem.

5 References

Altman, S. [@sama]. (2022, December 5). *ChatGPT launched on wednesday. Today it crossed 1 million users!* [Tweet]. Twitter. https://twitter.com/sama/status/1599668808285028353

Askari, A., Aliannejadi, M., Kanoulas, E., & Verberne, S. (2023). *Generating Synthetic Documents for Cross-Encoder Re-Rankers: A Comparative Study of ChatGPT and Human Experts* (v1.0). https://github.com/arian-askari/ChatGPT-RetrievalQA

Auto-GPT. (n.d.). *GitHub—Significant-Gravitas/Auto-GPT: An experimental open-source attempt to make GPT-4 fully autonomous.* Retrieved May 13, 2023, from https://github.com/Significant-Gravitas/Auto-GPT

Biswas, S. (2023). ChatGPT and the Future of Medical Writing. *Radiology*, *307*(2), e223312. https://doi.org/10.1148/radiol.223312

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., … Amodei, D. (2020). *Language Models are Few-Shot Learners.* https://doi.org/10.48550/ARXIV.2005.14165

Desaire, H., Chua, A. E., Isom, M., Jarosova, R., & Hua, D. (2023). *ChatGPT or academic scientist? Distinguishing authorship with over 99% accuracy using off-the-shelf machine learning tools* (arXiv:2303.16352). arXiv. https://doi.org/10.48550/arXiv.2303.16352

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. https://doi.org/10.48550/ARXIV.1810.04805

Doughman, J., Khreich, W., El Gharib, M., Wiss, M., & Berjawi, Z. (2021). Gender Bias in Text: Origin, Taxonomy, and Implications. *Proceedings of the 3rd Workshop on Gender Bias in Natural Language Processing*, 34–44. https://doi.org/10.18653/v1/2021.gebnlp-1.5

Hellström, J. (2023). *ChatGPT dialogues generated during thesis work in systems science.* [Data set]. https://doi.org/10.6084/m9.figshare.22822091.v1

Jun, Y. (2023, April 11). *Exploring Creativity in Large Language Models: From GPT-2 to GPT-4*. Medium. https://towardsdatascience.com/exploring-creativity-in-large-language-models-from-gpt-2-to-gpt-4-1c2d1779be57

Knight, W. (2022, December 7). ChatGPT's Most Charming Trick Is Also Its Biggest Flaw. *Wired*. https://www.wired.com/story/openai-chatgpts-most-charming-trick-hides-its-biggest-flaw/

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., & Stoyanov, V. (2019). *RoBERTa: A Robustly Optimized BERT Pretraining Approach*. https://doi.org/10.48550/ARXIV.1907.11692

Md_gender_bias · Datasets at Hugging Face. (n.d.). Retrieved May 27, 2023, from https://huggingface.co/datasets/md_gender_bias

Nayak, P. (2019, October 25). *Understanding searches better than ever before*. Google. https://blog.google/products/search/search-language-understanding-bert/

Oates, B. J. (2006). *Researching Information Systems and Computing*. SAGE Publications.

Olsson, E. (2023, January 11). *Lärarens rädsla: Att AI-boten förbjuds i skolan*. https://skolvarlden.se/artiklar/lararens-radsla-att-ai-boten-forbjuds-i-skolan

OpenAI. (n.d.-a). *Introduction—OpenAI API*. Chat Plugins. Retrieved June 10, 2023, from https://platform.openai.com/docs/plugins/introduction

OpenAI. (n.d.-b). *Models—OpenAI API*. Retrieved April 7, 2023, from https://platform.openai.com/docs/models/codex

OpenAI. (n.d.-c). *Models—OpenAI API*. Retrieved May 13, 2023, from https://platform.openai.com/docs/models/gpt-3-5

OpenAI. (2023). *What is ChatGPT? | OpenAI Help Center*. https://help.openai.com/en/articles/6783457-what-is-chatgpt

Radford, A., Narasimhan, K., Salimans, T., & Sutskever, I. (2018). *Improving Language Understanding by Generative Pre-Training*.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language Models are Unsupervised Multitask Learners. *OpenAI Blog*, *1*(8).

Roose, K. (2022, December 5). The Brilliance and Weirdness of ChatGPT. *The New York Times*. https://www.nytimes.com/2022/12/05/technology/chatgpt-ai-twitter.html

Roose, K. (2023, January 12). Don't Ban ChatGPT in Schools. Teach With It. *The New York Times*. https://www.nytimes.com/2023/01/12/technology/chatgpt-schools-teachers.html

Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). *DistilBERT, a distilled version of BERT: Smaller, faster, cheaper and lighter*. https://doi.org/10.48550/ARXIV.1910.01108

Sennrich, R., Haddow, B., & Birch, A. (2015). *Neural Machine Translation of Rare Words with Subword Units*. https://doi.org/10.48550/ARXIV.1508.07909

The Moscow Times. (2023, February 2). *Russian Student Allowed to Keep Diploma for ChatGPT-Written Thesis—The Moscow Times*. https://www.themoscowtimes.com/2023/02/02/russian-student-allowed-to-keep-diploma-for-chatgpt-written-thesis-a80125

Thomas H. Ptacek [@tqbf]. (2022, December 2). *I'm sorry, I simply cannot be cynical about a technology that can accomplish this. Https://t.co/yjlY72eZ0m* [Tweet]. Twitter. https://twitter.com/tqbf/status/1598513757805858820

Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., Rodriguez, A., Joulin, A., Grave, E., & Lample, G. (2023). *LLaMA: Open and Efficient Foundation Language Models*. https://doi.org/10.48550/ARXIV.2302.13971

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). *Attention Is All You Need*. https://doi.org/10.48550/ARXIV.1706.03762

Wiggers, K. (2023, January 5). *As NYC public schools block ChatGPT, OpenAI says it's working on 'mitigations' to help spot ChatGPT-generated text | TechCrunch*. https://techcrunch.com/2023/01/05/as-nyc-public-schools-block-chatgpt-openai-says-its-working-on-mitigations-to-help-spot-chatgpt-generated-text/

Zhadan, A. [@biblikz]. (2023, January 31). *Защитил диплом, написанный ChatGPT. Поделюсь, как организовал процесс, что услышал от людей о получившемся тексте и почему должен чизкейк. Вышло ненапряжно и прикольно! Https://t.co/soFdcTcynW* [Tweet]. Twitter. https://twitter.com/biblikz/status/1620451262822252544

Zheng, H., & Zhan, H. (2023). ChatGPT in Scientific Writing: A Cautionary Tale. *The American Journal of Medicine*, S0002934323001596. https://doi.org/10.1016/j.amjmed.2023.02.011