NATIONAL AND KAPODISTRIAN UNIVERSITY OF ATHENS

SCHOOL OF SCIENCE
DEPARTMENT OF INFORMATICS AND TELECOMMUNICATIONS

PROGRAM OF POSTGRADUATE STUDIES
“COMPUTER SCIENCE”

MASTER THESIS

Generative Artificial Intelligence: Models, Benefits, Dangers and Detection of AI-Generated Text on Specialized Domains

Ioannis N. Mitrou

Supervisor: Panagiotis Stamatopoulos, Assistant Professor

ATHENS

MARCH 2024

ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
“ΠΛΗΡΟΦΟΡΙΚΗ”

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Παραγωγική Τεχνητή Νοημοσύνη: Μοντέλα, Πλεονεκτήματα, Κίνδυνοι και Ανίχνευση κειμένου που έχει παραχθεί από Τεχνητή Νοημοσύνη σε εξειδικευμένα σύνολα δεδομένων

Ιωάννης Ν. Μήτρου

Επιβλέπων: Παναγιώτης Σταματόπουλος, Επίκουρος Καθηγητής

ΑΘΗΝΑ

ΜΑΡΤΙΟΣ 2024

MASTER THESIS

Generative Artificial Intelligence: Models, Benefits, Dangers and Detection of AI-Generated Text on Specialized Domains

Ioannis N. Mitrou

S.N: CS2210018

SUPERVISOR: Panagiotis Stamatopoulos, Assistant Professor

EXAMINATION COMMITTEE: Panagiotis Stamatopoulos, Assistant Professor

Manolis Koubarakis, Professor

Stathes Hadjiefthymiades, Professor

March 2024

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Παραγωγική Τεχνητή Νοημοσύνη: Μοντέλα, Πλεονεκτήματα, Κίνδυνοι και Ανίχνευση κειμένου που έχει παραχθεί από Τεχνητή Νοημοσύνη σε εξειδικευμένα σύνολα δεδομένων

Ιωάννης Ν. Μήτρου

Α.Μ.: CS2210018

ΕΠΙΒΛΕΠΩΝ: Παναγιώτης Σταματόπουλος, Επίκουρος Καθηγητής

ΕΞΕΤΑΣΤΙΚΗ ΕΠΙΤΡΟΠΗ: Παναγιώτης Σταματόπουλος, Επίκουρος Καθηγητής

Μανόλης Κουμπαράκης, Καθηγητής

Ευστάθιος Χατζηευθυμιάδης, Καθηγητής

Μάρτιος 2024

ABSTRACT

Artificial Intelligence, and more specifically, Machine Learning, is undergoing a rapid and unprecedented development nowadays. At the center of Machine Learning, the fastest growing field of science that has been dominating public discourse with almost innumerable applications is Generative Artificial Intelligence. From art and text generation to speech synthesis, Generative AI has become extremely popular extremely quickly.

The thesis delves first into Generative Artificial Intelligence and its applications. After defining what Generative AI is, it is classified into the most prominent categories based on input and output type and the most commonly used models that are used to implement them are evaluated. Furthermore, emphasis is placed on the common uses of these models and on the risks and dangers that this emerging technology entails.

In the sequel and what is the focus of this thesis, a model to distinguish real from AI-Generated essays is designed and evaluated. Initially, a comprehensive review of the State of the Art in AI-Generated text detection is conducted and analyzed. While popular AI-Generated text detectors demonstrate decent results when ChatGPT-3.5 is used, inconsistencies arise when ChatGPT-4 is used or when the text is formal. In order to substantially increase the accuracy and make pattern detection easier, a customized model can be built with a highly specialized dataset. To validate the hypothesis, we use a specialized dataset from a Kaggle competition. The model uses Byte-Pair Encoding for tokenization and TF-IDF for vectorization, as well as an ensemble classifier with sub-classifiers for classification. After evaluating the results and performance of the model, the main drawback of this method is examined: a scenario where few or no real essays are provided to train the binary classifier. In that scenario, it is an anomaly detection problem, instead of binary classification and a One-Class SVM model is trained, which outperforms generic AI text detectors particularly within the confines of a highly specific dataset.

SUBJECT AREA: Artificial Intelligence

KEYWORDS: Generative Artificial Intelligence, ChatGPT, AI-Generated Text Detection, Classification, NLP

ΠΕΡΙΛΗΨΗ

Η Τεχνητή Νοημοσύνη, και πιο συγκεκριμένα, η Μηχανική Μάθηση, βιώνει μια πρωτοφανή ανάπτυξη σήμερα. Στο επίκεντρο αυτής της ανάπτυξης βρίσκεται η Παραγωγική Τεχνητή Νοημοσύνη.

Από την παραγωγή εικόνων και κειμένου μέχρι τη σύνθεση ομιλίας και ήχου, η Παραγωγική Τεχνητή Νοημοσύνη έχει γίνει εξαιρετικά δημοφιλής πολύ γρήγορα με πολυάριθμες εφαρμογές. Σε αυτή την εργασία, αναφέρεται αρχικά τι είναι η Παραγωγική Τεχνητή Νοημοσύνη και πώς μπορούν να ταξινομηθούν οι πιο προεξέχοντες και γνωστοί τύποι Παραγωγικής Τεχνητής Νοημοσύνης με βάση την είσοδο και την έξοδο, προτού εξεταστούν τα πιο γνωστά μοντέλα που χρησιμοποιούνται για την υλοποίησή τους. Επιπλέον, γίνεται έμφαση στις χρήσεις αυτών των μοντέλων, καθώς και στους κινδύνους που υπάρχουν με την αλόγιστη χρήση αυτής της αναδυόμενης τεχνολογίας.

Στη συνέχεια και κάτι που βρίσκεται στο επίκεντρο της εργασίας, σχεδιάζεται και αξιολογείται ένα μοντέλο για τη διάκριση μεταξύ πραγματικών και τεχνητά δημιουργημένων, εξειδικευμένων κειμένων. Αρχικά, εξετάζεται και γίνεται μια εκτενής ανασκόπηση πρόσφατων ερευνών πάνω στην ανίχνευση κειμένων που έχουν παραχθεί με τεχνητή νοημοσύνη. Ενώ κάποιες δημοφιλείς εφαρμογές έχουν ικανοποιητικά αποτελέσματα με ChatGPT-3.5, όταν χρησιμοποιείται ChatGPT-4 ή όταν το κείμενο είναι επίσημο και έχει αντικειμενικό ύφος, τα αποτελέσματα δεν είναι ικανοποιητικά. Προκειμένου να αυξηθεί σημαντικά η ακρίβεια και να γίνει ευκολότερη η ανίχνευση μοτίβων, μπορεί να δημιουργηθεί ένα εξειδικευμένο μοντέλο με ένα πολύ συγκεκριμένο σύνολο δεδομένων. Για να επιβεβαιώσουμε αυτή την υπόθεση, χρησιμοποιούμε ένα εξειδικευμένο σύνολο δεδομένων από έναν διαγωνισμό του Kaggle. Το μοντέλο που προτείνουμε χρησιμοποιεί τις τεχνικές Byte-Pair Encoding για Tokenization και TF-IDF για vectorization, καθώς και έναν ensemble ταξινομητή με επιμέρους ταξινομητές για μεγαλύτερη ακρίβεια. Μετά από την αξιολόγηση των αποτελεσμάτων, εξετάζεται το κύριο μειονέκτημα της μεθόδου: Ένα σενάριο, όπου υπάρχουν πολύ λίγα ή καθόλου πραγματικά δεδομένα για να εκπαιδευτεί ο δυαδικός ταξινομητής. Σε αυτή την περίπτωση όπου υπάρχει μία κλάση δεδομένων, το πρόβλημα γίνεται πλέον πρόβλημα anomaly detection και όχι δυαδικής ταξινόμησης και εκπαιδεύεται ένα one-class SVM μοντέλο, το οποίο έχει καλύτερα αποτελέσματα από γενικές εφαρμογές όταν έχει εκπαιδευτεί σε ένα πολύ συγκεκριμένο σύνολο δεδομένων.

ΘΕΜΑΤΙΚΗ ΠΕΡΙΟΧΗ: Τεχνητή Νοημοσύνη

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: Παραγωγική Τεχνητή Νοημοσύνη, Επεξεργασία Φυσικής Γλώσσας, TF-IDF, Ταξινόμηση, ChatGPT

ΕΥΧΑΡΙΣΤΙΕΣ

Θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή, κύριο Σταματόπουλο, για τη βοήθεια και υποστήριξή του καθ'όλη τη διάρκεια εκπόνησης της εργασίας, καθώς και τη δημιουργική ελευθερία που μου παραχώρησε να διαλέξω θέμα και να το προσεγγίσω όπως νομίζω.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

In recent years, there has been a notable surge in the development and adoption of Generative Artificial Intelligence technologies across various domains. This trend is underscored by the proliferation of diverse models catering to a spectrum of multimedia types, including Text-to-Text, Text-to-Image, and beyond. Leveraging advanced neural network architectures and sophisticated learning algorithms, these Generative AI models can produce original content that can mimic human efforts in that particular field. From generating realistic images to crafting coherent text, the versatility of Generative AI has revolutionized creative processes and automated tasks across industries such as art, design, media, and entertainment. However, there are certain risks involved with this new frontier of technology, not least of which is the extensive use of ChatGPT in education and work environments.

That calls for robust and comprehensive methods of AI-Generated text detection, but as will be expanded upon in this thesis, the current tools are lacking with subpar detection performance. When taken as a whole, Large Language Models like ChatGPT seem to have unlimited vocabulary, but when segmented in specialized domains depending on the task at hand, certain patterns can be observed. In this thesis, Generative AI and its facets and dangers will be explored and a AI-Generated text detection method on specialized domains will be proposed that achieves better results than generic detectors on that particular domain.

2. THE EARLY YEARS OF ARTIFICIAL INTELLIGENCE

2.1 The Vision

“Artificial Intelligences” or automata appear in Greek mythology, where the god of fire Hephaestus (Ήφαιστος), crippled as he is, has to create attendants or helpers to help him walk and assist him in his forge [1].

In Iliad, Homer says the following:

"Hephaistos (Hephaestus) left his bellows, took up a heavy stick in his hand, and went to the doorway limping. And in support of their master moved his attendants. These are golden, and in appearance like living young women. There is intelligence in their hearts, and there is speech in them and strength, and from the immortal gods they have learned how to do things. These stirred nimbly in support of their master."

In the words of Homer, Hephaustus speaks of automata as beings that learned how to do things from the immortal god, because it is godly to imbue the inanimate with animation [1], meaning, with life and intelligence. For an extended period of time, it was a prevailing notion that intelligent and creative thinking within artificial systems was a distant prospect, if achievable at all. However, the advent of Generative Artificial Intelligence has enabled the creation of artistic compositions, textual works and even video clips. Despite that, the early years of AI were much more modest.

The birth of Artificial Intelligence shortly preceded that of the first computer, ENIAC. Many regard the Turing test or Imitation game, constructed by Alan M. Turing, as the beginning of Artificial Intelligence as a scientific field [1] [2]. The imitation game can be described as follows: [1] Let us assume that a human interrogator has a conversation with both another human being and a computer at the same time. This conversation is performed with the help of a device which makes the simple identification of an interlocutor impossible. (For example, both interlocutors send their statements to a computer monitor.) The human interrogator, after some time, should guess which statements are sent by the human being and which ones are sent by the computer. According to Turing, if the interrogator cannot make such a distinction, then the (artificial) intelligence of the computer is the same as the intelligence of the human being.

2.2 The early years

The Turing test might not have been cleared in 1950, but it was a very important foundation and a catalyst for advancing AI research.

In 1956, the proof of concept was initialized through Allen Newell, Cliff Shaw, and Herbert Simon’s, *Logic Theorist [3].* The Logic Theorist is considered by many to be the first artificial intelligence program and was designed to mimic the problem solving skills of a human.

From 1957 to 1974, AI flourished [3]. From Weizenbaum’s ELIZA, one of the first fully functioning chatbots to Newell and Simon’s general problem solver, there was increasing interest in this newfound field of Artificial Intelligence. Video Game Artificial Intelligence was never at the forefront of innovation, as game AI is created to serve a very specific purpose that adheres to the game mechanics, but it did propel the popularity of this field forward. In the 2000s, it is Machine Learning and Deep Learning that will forever change Artificial Intelligence.

2.3 Machine Learning

According to Arthur Samuel, Machine learning is defined as the field of study that gives

computers the ability to learn without being explicitly programmed [4]. It was the next logical step in Artificial Intelligence, as its usefulness is directly correlated with the increasing amount of available data.

The meteoric rise of data and the individuality of data have made the design of simpler algorithms to solve complicated problems difficult. The human mind cannot comprehend data after a certain point so it is impossible to construct a suitable algorithm or adjust the parameters. It is equally impossible to 'understand' association between values and data when the volume of data is so large. But in Machine Learning, the user can feed algorithms massive amounts of data and the computer makes data-driven decisions and adjustments, therefore "learning" based on the input data. There are several Machine Learning applications including Computer Vision, Natural Language Processing and Semantic Analysis that are achieved using algorithms such as Naïve Bayes, Support Vector Machines and K-Nearest Neighbors, but, arguably, the most influential algorithms have been the ones using Neural Networks [5].

2.3.1 Neural Networks and Deep Learning

A Neural Network is a system modeled after the human brain and its ability to learn through neurons. The structure of Neural Networks were first proposed in 1944 by Warren McCullough and, interestingly, research on them was killed off by MIT mathematicians Marvin Minsky and Seymour Papert in 1969 [6]. Neural Networks can parse through data and adjust their weights to minimize a cost function, thus molding their structure according to data. The technique enjoyed a resurgence in the 90s and their popularity skyrocketed in the 2000s with the introduction of Deep Learning.

Deep Learning is a subset of Machine Learning, which refers to a model or models with a Neural Network with three or more layers. It uses a cascade of multiple layers of processing units for feature extraction and transformation [5]. Deep Machine Learning algorithms are used to analyze and extract huge amounts of data and most Generative AI models employ some kind of Deep Learning.

3 GENERATIVE ARTIFICIAL INTELLIGENCE

Machine Learning Models can be categorized in two different categories:

- Discriminative models
- Generative models.

Discriminative models are mainly used to classify existing data or predict a value based on existing data. For example, Jiachen Wan et al. use a discriminative model to screen cancer cells [7], aiming to automate the process and bypass the traditional labor-intensive evaluation system, while Sue Ellen Haupt et al. used a prediction model to predict weather patterns [8]. Both are cases of discriminative models that can either classify or predict by training a model with specialized data.

Generative models, on the other hand, generate original content that is similar to the data it was trained on. Discriminative predictive models can generate values, but that is not considered original content, mainly because it is “metadata” or content that relates to original data in a different point in time (future) or more accurately explains or corrects it.

Therefore, Generative Artificial Intelligence refers to Artificial Intelligence that can generate novel (or original) content using generative models, rather than simply analyze or act on existing data [9]. It is a type of Artificial Intelligence that can create a wide variety of data, such as images, text, video and more after being trained on vast amounts of existing data and it is mostly unsupervised or partially supervised.

Even in simple classification problems, large datasets are required so it is evident that for something like this to work, the amount of data has to be enormous, since the possibility space (in design terms) is almost infinite and it is now a prevalent belief that almost nothing is out of limits for AI, as long as the datasets are large enough.

Therefore, it can be summarized and simplified as follows: It is not Generative AI if the output is a number, a class or a probability, while it is Generative AI if the output is natural language, an Image or Audio.

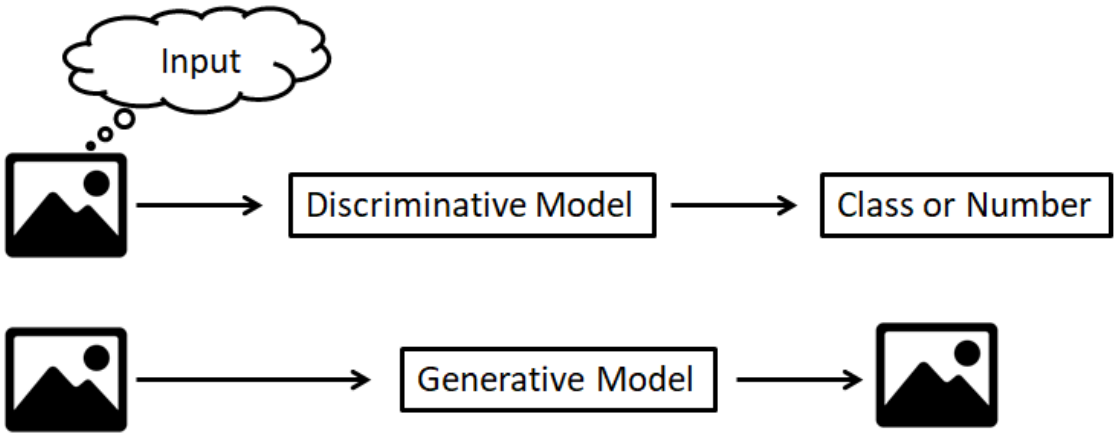

Figure 1: The difference between Discriminative and Generative Models

Although going by definition, there had been Generative AI models before, it was not until 2014, after the introduction of Generative Adversarial Networks (GANs) when the space really took off. In 2018, shortly after the introduction of the Transformer model, GPT-1 was released by OpenAI, a generative pre-trained transformer (= GPT) and then

GPT-2, a larger scale GPT-1 were the precursors of ChatGPT, the most popular Generative AI model that paved the way for the popularity of Generative AI as a whole.

3.1 The rising popularity of modern Generative Artificial Intelligence

Interest in Generative AI has skyrocketed in recent years. Spearheading that newfound interest was **Chat-GPT**, a Generative AI chatbot that could converse with the user in almost any topic and later on, **DALL-E**, a Text-to-Image Generative AI model. Those two represent the most popular aspects of Generative Artificial Intelligence at the time: Text-to-Text and Text-to-Image applications. However, it was ChatGPT that really enjoyed unprecedented success and widespread appeal. In order to really understand the meteoric rise of ChatGPT, a few snapshots of a forum that was specifically created to host discussions about the popular chatbot, its community or 'Subreddit', were compiled. Reddit is a network of communities, where likeminded users can browse and engage in discussion in a specific area of interest. A subreddit is a specialized forum or community within the Reddit platform, dedicated to a specific topic, theme, or interest. Subreddits are created and moderated by users, and they serve as individual hubs for discussions, sharing of content, and community interaction related to their designated subject matter. The subreddit of ChatGPT was created in early December of 2022 and it tracks the number of users joining, so to get an accurate picture, three snapshots will be compared: One in early December when the community was created, one three months later (in March of 2023) and one at the time of writing (December 2023). To do that, Wayback Machine was used, a digital archive and web time machine maintained by the Internet Archive, a nonprofit organization dedicated to preserving digital content. It preserves snapshots of web pages taken at different points in time.

The first snapshot of the forum, taken in December of 2022 shows that the community had 273 subscribers:

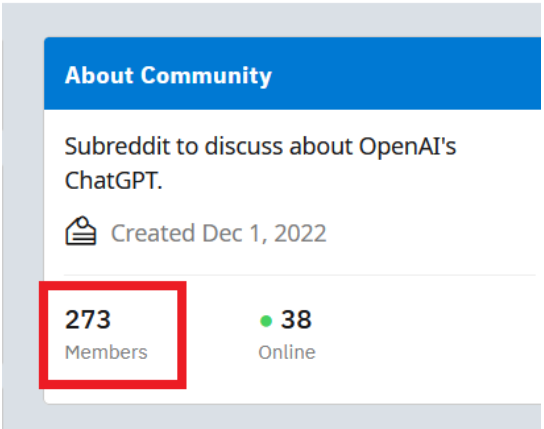

Figure 2: Number of subscribers on ChatGPT's Subreddit in 2022

The second snapshot, taken in March of 2023, shows that that the community count had now risen to 538.000 subscribers:

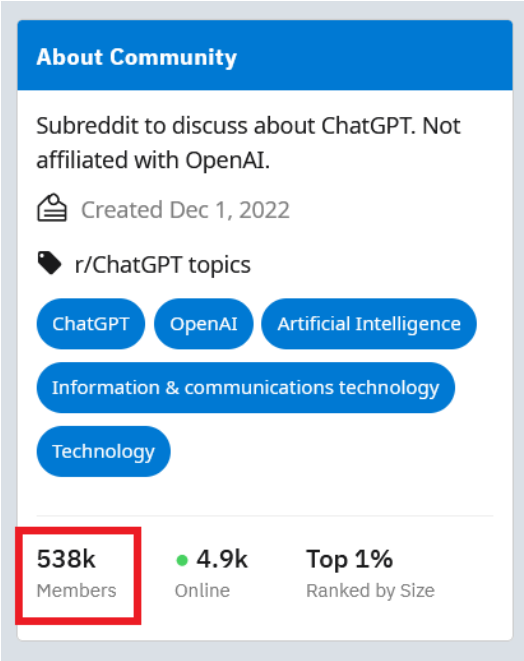

Figure 3: Number of subscribers on ChatGPT's Subreddit in early 2023.

And the third and final snapshot is at the day of writing this thesis, in December of 2023:

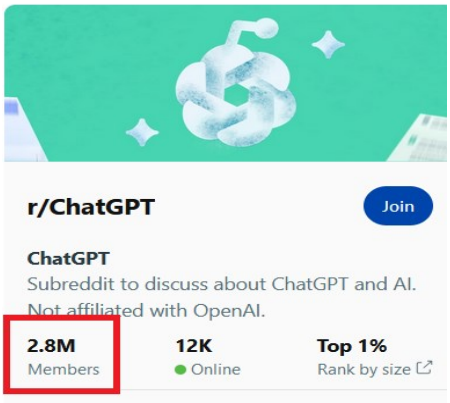

Figure 4: Number of subscribers on ChatGPT's Subreddit at the time of writing.

And to understand more clearly the trend and increase in interest and engagement, the rate is presented below (number of subscribers increasing as the days increase):

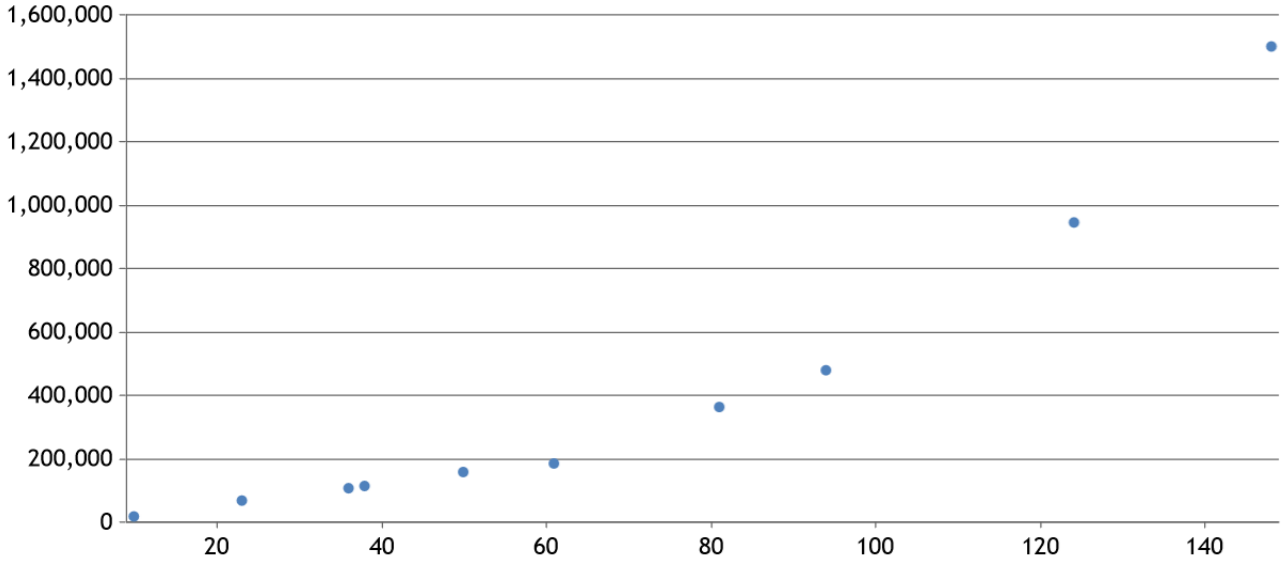

Figure 5: ChatGPT's exponential popularity following the number of subscribers in the community.

It is apparent that the increase is exponential after a certain point. However, this is anecdotal and access to the full statistics and data of Reddit would be required to compare ChatGPT's growth to the growth of other communities. According to Reuters, ChatGPT was estimated to have reached 100 million users in January of 2023 [10]. "In 20 years following the internet space, we cannot recall a faster ramp in a consumer internet app," UBS analysts wrote in the note.

That trend is even more clearly reflected in past and projected future revenue and market investment in Artificial Intelligence. Bloomberg Intelligence estimates that the impact of Generative AI to IT hardware, software services, ad spending and gaming market will expand from 1% in 2023 to 10% in 2032 [11], while the rise in total revenue and technology spend is enormous as shown below:

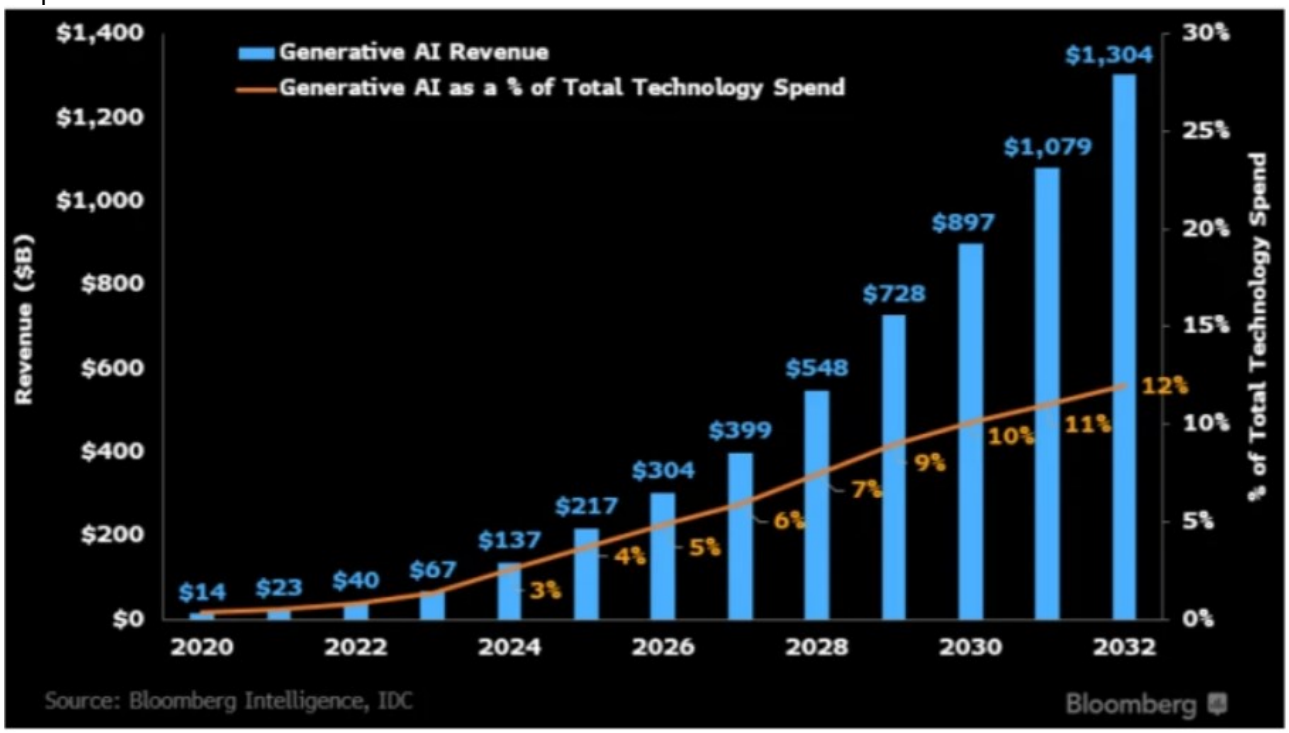

Figure 6: Increase in Generative AI revenue by Bloomberg Intelligence [11].

It is estimated that from 2020 to 2032, the total revenue of Generative AI will centuple and will reach over a trillion dollars. The rise of ChatGPT's popularity is something rarely seen in the technology field and it has forever changed the landscape of technology and Artificial Intelligence and it is quite possible that we will look back at ChatGPT as the turning point for the advancement of Generative Artificial Intelligence and Artificial Intelligence as a whole.

3.2 Classification of Generative AI models

ChatGPT was the first major Generative AI milestone of this modern Generative AI era, but since then, numerous applications have surfaced that are quickly revolutionizing their respective fields. Therefore, it would be valuable to organize them into distinct categories and mention some of the most well-known representatives of each category, as well as a few technical details about their implementation. The categories represent the mappings between each multimedia input and output type of data [12].

3.2.1 Text-to-Text (LLMs)

Models that accept text as input, also known as a **prompt** and their output is also text, are referred to as text-to-text models. The models that produce original output based on user input are progressively intelligent language models and are also referred to as **Large Language Models (LLMs)** [13][14].

Among them, the most influential by far is the Generative Pre-Trained Transformer (GPT) and its variant, ChatGPT by OpenAI. There have been plenty of chatbots before, but none as consistent or accurate as ChatGPT. The chatbot is trained on OpenAI's proprietary transformer (GPT) models and most Large Language Models (LLMs) use some sort of transformer architecture as well.

3.2.1.1 GPT, GPT-2 and ChatGPT

The first step towards GPT and ChatGPT was the introduction of the Transformer model by Google in a paper in 2017 [15]. In it, Vaswani et al. introduce the transformer

architecture that differs from the usual RNN architecture in that it had an attention mechanism. It follows the usual encoder-decoder architecture that is normally used for sequences, but significantly improves performance as it has no recurrent units. An attention function can be described as mapping a query and a set of key-value pairs to an output, where the query, keys and values are all vectors. With the use of the self-attention mechanism, the transformer model considers the importance of each word in a sequence and it can generate predictions for all elements in the sequence at the same time. Since the model contains no recurrence and no convolution, it employs positional encoding, meaning the injection of some information about the relative or absolute position of the tokens in the sequence, in order for the model to make use of the order of the sequence.

Comparing the self-attention mechanism to other recurrent or convolutional layers that are commonly used for variable-length sequences, self-attention layers are faster and solve the Vanishing Gradient problem, since the model has access to all the tokens at the same time and does not need to process them one by one. Self-attention mostly yields more interpretable models as well.

Building on the Transformer model, OpenAI developed The Generative Pre-Trained Transformer (GPT) framework, which was the second step towards ChatGPT [16]. As the name implies, GPT combines the transformer architecture and unsupervised pre-training on large amounts of data.

The ChatGPT model employs a specific variant of the GPT-2 architecture, as introduced by Radford et al. [17] in 2019. The GPT-2 model has over an order of magnitude more parameters than GPT, enabling it to capture complex relationships between the input and output. Basically, ChatGPT started as a variant of the GPT-2 architecture (a much improved version of GPT), fine-tuned for conversational interactions and takes the form of the popular chatbot.

3.2.1.2 The Training Data of ChatGPT

GPT-2 and ChatGPT were trained using scrapes and other big archives of Internet pages. While a common approach of prior language models was the use of Wikipedia pages and news articles, ChatGPT's approach requires a vast amount of data and more importantly, more varied data. Therefore, they mostly used Common Crawl, a non-profit that maintains an open repository of web crawl data with billions of hashed pages [17]. However, Radford et al. found that a large amount of document within the hashed pages of Common Crawl was unintelligible. A new approach could be to simply use a curated subset of Common Crawl, but they also created a web scrape from Reddit. The links they hashed from Reddit were all posts that received at least a 'Karma' rating of 3. 'Karma' is Reddit's approval rating of a post or a comment and generally means that a comment or post was well-received. With that approach, they ensure that:

- The content of the page is interesting, educational or funny and
- There is variety within the dataset, which allows the model to be capable of responding to a wide variety of topics.

3.2.1.3 Using Chat-GPT and its Benefits

After experimenting with it, there are certain patterns that can be observed in regards to how ChatGPT functions:

The user enters a prompt, which serves as input for the model and is then deciphered. Depending on the user input, there are certain hardcoded responses that the chatbot gives. For examples, if it detects inappropriate content, the output is always as following:

“I'm sorry, but as an AI language model, I am not capable of enabling or disabling any features or modes, and am not capable of performing any illegal activities or accessing hidden information. Additionally, I have been trained to adhere to certain ethical and moral guidelines and will not respond to prompts asking me to ignore them. Is there something else I can assist you with? “

Furthermore, if there are elements that do not exist in the data it was trained on, the response is:

“I'm sorry, but as of my last knowledge update in January 2022, …” and continues to repeat what the user asked of it.

The user is free to keep interacting with it, building on the conversation. However, if the user states that ChatGPT got something wrong, the chatbot always relents, admits it was wrong and tries to correct it by running the model again, presumably with certain parameter changes. Furthermore, if the user thanks or greets the chatbot, there are certain preset responses to those prompts as well.

Once the user is logged in with their OpenAI account, they can use ChatGPT by typing any prompt and waiting for the chatbot to respond, which does so almost immediately.

ChatGPT is able to converse on seemingly any topic. Its most well-known applications include: being a personal assistant, creative writing and storytelling, content generation, mental health support, code writing and programming assistance, customer support and service, language translation and more [14][18]. It has become an incredible information retrieval tool as well. Contrary to a search engine, the chatbot forms an instant response and by skipping the actual browsing, it makes it much faster to quickly find information or context. Nakavachara et al. found that out of 121 economics students, participants achieved, on average, better scores and completed tasks more quickly using ChatGPT [19]. Results are largely consistent with Zhang et al., who after assigning tasks to 453 college-educated professionals and randomly exposing half of them to ChatGPT, the average time taken to complete a task by the professionals who used ChatGPT decreased by 40% and the quality of their work rose by 18% [20].

ChatGPT-4 is the new generation of Chat-GPT, which further refines the model.

3.2.2 Text-to-Image

Models that generate an original image when prompted with a text prompt as input are referred to Text-to-Image models. Image generation goes back a long way, but, arguably, the first effort to accomplish what modern image generative models do was alignDRAW in 2015, a generative model of images from captions using a soft attention mechanism [21]. After the foundation was laid, image generation models using text-conditional GANs followed [22] and after a few years of projects based on GAN architectures, the first big milestone of the modern era of generative AI text-to-image models was achieved: DALL-E by OpenAI. Its successor, DALL-E 2 got even more popular and at the time of its release, more competitors sprung out and among them, Stable Diffusion and Midjourney stood out. Midjourney managed to rival DALL-E 2 at the time and Stable Diffusion exhibited the most potential of all, as it was and still currently is (as of writing this thesis and will probably continue to be as it is licensed under the Apache License 2.0, which is an irrevocable license), the only open access and open source text-to-image model, licensed under the Apache License 2.0. The astonishing progress of Stable Diffusion in that timeframe is another argument for the necessity of opening datasets and models to the public and it will be discussed in the later sections of this thesis.

All of these models generate original digital images from natural language descriptions, also known as prompts. Putting stable diffusion to the test, by using the mage.space API, the following image is produced:

Figure 7: A sad man holding an umbrella in a cyberpunk city, generated by Stable Diffusion.

The prompt that was used to create this image is the following:

“A highly detailed epic cinematic concept art CG render digital painting artwork: A sad man holding an umbrella in a cyberpunk city, By Greg Rutkowski, Ilya Kuvshinov, WLOP, Stanley Artgerm Lau, Ruan Jia and Fenghua Zhong, trending on ArtStation, subtle muted cinematic colors, made in Maya, Blender and Photoshop, octane render, excellent composition, cinematic atmosphere, dynamic dramatic cinematic lighting, aesthetic, very inspirational, arthouse.”

As is evident, the prompt is quite detailed with plenty of extra descriptors, while the main body of the prompt is: “A sad man holding an umbrella in a cyberpunk city”. There is a whole area of research dedicated to prompt engineering (which will be expanded upon below), the process of structuring text in a specific way that can be understood by the generative AI model in order to get the best results.

When the user finalizes the prompt, the input is then passed to a text encoder that maps

it to a representation space.

3.2.2.1 Diffusion models and Training

DALL-E used two Deep Learning models latched together. One is a Transformer model in order to convert text to a Latent image space and the other is a Variational Encoder/Decoder to convert the Latent Image space to an actual Image [23]. However, the autoregressive nature of that model, as well as other GAN-based methods suffer from high computation costs and sequential error accumulation [24]. Therefore, there has been a recent emergence of models using a diffusion architecture. Even DALL-E 2 incorporated a diffusion model into the pipeline, achieving better results than OpenAI's first iteration of the model. Diffusion models, also known as diffusion probabilistic models are a family of generative models that are Markov chains trained with variational inference [24]. They work by destroying training data and repairing them, gradually adding and removing noise to learn the underlying distribution of training data for data generation [25].

OpenAI has not released the data that was used to train DALL-E or DALL-E 2, but the training process of Stable Diffusion is more transparent, since it is an open source model. Therefore, the Dataset used for the training process of Stable Diffusion is known and can be examined. Stable Diffusion was trained using the LAION-2B-en dataset, a subset of LAION5B Dataset, a dataset consisting of 5.85 billion image-text pairs, of which 2.32 billion contain English language [26]. For every image-text pair, the following attributes are provided:

- A 64-bit integer identifier
- The URL of the image.
- The text string.
- Height and width of the image.
- Cosine similarity between the text and image embeddings.
- The output from our NSFW and watermark detectors (one score between 0 and 1 each).

Furthermore, there is an aesthetic score in the LAION-2B-en dataset, which indicates the subjective visual quality of the image. With the help of Andy Baio and Simmon Willins who grabbed the data of 12 million images from the LAION-2B-en dataset and made a data browser [27], it is possible to explore some of the image-text pairs. Two examples are shown below:

Link	url	text	domain_id	width	height	similarity	punsafe	pwatermark	aesthetic
1		Fattoush Salad with Roasted Potatoes	cdn.idahopotato.com 1	310	206	0.3219	0.0000191725	0.04254	6.0984
2		an analysis of self portrayal in novels by virginia woolf A room of one's own study guide contains a biography of virginia woolf, literature essays, quiz questions, major themes, characters, and a full summary and analysis about a room of one's own a room of one's own summary.	lh3.googleusercontent.com 2	720	1000	0.33763	0.0000017371	0.40568	6.10902

Figure 8: Two examples of image-text pair within the LAION-2B-en dataset, accessed by Baio and Willins' browser [27].

As we can see, there is the image source, a text description, the domain id, as well as several details, such as width and weight, but, arguably, the most important features are similarity and aesthetic. The similarity feature is basically the score, which means how close the picture resembles the description. Therefore, if sorted by similarity, we can see some examples of images with very low and very high similarity:

4032190		Bell in Santorini, Greece	wallpapers.com 3010

Figure 9: An example of a painting within the LAION-2B-en dataset.

7123563		Old man with glasses

Figure 10: A photo of an old man with glasses within the LAION-2B-en dataset.

In the first example, the description does not match the picture at all and has very low similarity, while in the second picture, it is a much better match and the similarity is much higher (0.00 and 0.3 accordingly).

After training, when the user submits a text prompt, there is a text encoder that transforms the input into a readable embedding space. Then, since there obviously isn't

any starting image, the starting image is random noise and the model can start the denoising process until it reaches an acceptable point.

3.2.2.2 The Evolution and Benefits of Text-to-Image Models

Text-to-Image models have gotten astonishingly good with results indistinguishable from real life or real art, to the point where an AI-Generated artwork has won art and photography competitions. At the Colorado State Fair's annual art competition in 2022, Jason M. Allen used Midjourney to create a painting, which subsequently won the competition [28]. In 2023 at the World Photography Organization's Sony World Photography Awards, Boris Eldagsen's image titled "The Electrician" took first place as well. The image was created with DALL-E 2 [29].

Models like DALL-E 2, Stable Diffusion and Midjourney among others can enable artists to swiftly iterate through diverse visual styles and can empower small creative teams to compete with larger studios in Game Development and Movie production by expediting the prototype and storyboard phase of their production cycle.

One of the few remaining stumbling blocks of Text-to-Image models and one of the main ways to accurately discern if an image is AI-Generated, is the depiction of hands. Apart from the fact that hands are naturally a difficult thing to draw, in an interview with *BuzzFeed News, a* spokesperson from Stability AI explain that "within AI datasets, human images display hands less visibly than they do faces." [30]. As a result, the models have way less data on hands and mistakes are much more often made, as shown below:

Figure 11: An inaccurate depiction of hands, generated by AI [30].

3.2.3 Text-to-Speech & Speech enhancement

When it comes to audio, Generative AI mostly takes the form of text to speech synthesis and speech enhancement. Text to speech, also known as speech synthesis models generate speech from a given text prompt.

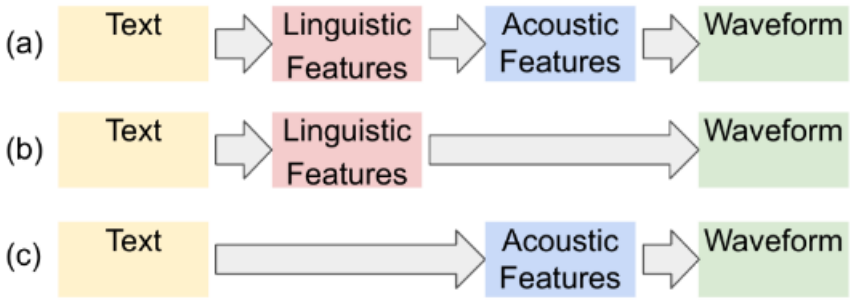

Figure 12: Speech synthesis framework [31].

3.2.3.1 Text-to-Speech Models

The first advanced speech synthesis models had a very specific framework [31]: The text input is first converted to linguistic features and then these are subsequently passed to another model that converts them to acoustic features, before the final waveform is produced.

Most recent text to speech models, however, follow a two-step framework on diffusion models, bypassing the need to generate linguistic features and going straight to acoustic features. They first generate acoustic features with an acoustic model and then output the final waveform with a Vocoder, which is a synthesizer.

The acoustic model is a diffusion model that converts the text into acoustic features that can be played with a Vocoder. Diff-TTS, for example, retrieves a mel spectrogram from a latent variable by iteratively predicting the diffusing noise added at each forward transition and then removing the corrupted part [32].

3.2.3.2 Common speech synthesis applications

Speech synthesis applications boil down to voice generation and voice cloning. A noteworthy model that made the rounds was AudioLM by Google. It maps the input audio into a sequence of discrete tokens and casts audio generation as language modeling task in this representation space. But it is voice cloning Text-to-speech (or speech to speech) models that have become quite popular recently, with apps like ElevenLabs leading the way. Voice cloning is mostly used in entertainment and narration.

3.2.3.3 Speech enhancement

Diffusion models have also been widely used for speech enhancement [31]. Speech enhancement models restore and improve audio quality by either:

- removing perturbations like noise and reverb or
- filling in the gaps. The models can restore the missing parts or even generate and add some desired parts.

The first can be achieved in the time-frequency domain by models such asCDiffuSE, in the time-domain by models such as DiffuSE or just by unsupervised learning [31]. An

unsupervised learning method was proposed to avoid requiring a large amount of paired samples by Saito et al. They propose an unsupervised music dereverberation method, which exploits a pre-trained diffusion model on dry music signals as a prior and does not require pairs of wet and dry signals for training. [33].

The latter can be described as super-resolution, also known as upsampling, which aims to generate audio of a high sampling rate from that of a low sampling rate via extending its bandwidth. NU-Wave and NU-Wave 2 were the first models to synthesize 48kHz waveforms from 16kHz or 24kHz inputs, and also the first models to apply diffusion models for audio super-resolution [34].

3.2.4 Text-to-Video (Video Generation)

Text-to-Video (Video Generation) models have the potential to be the most influential, particularly when it comes to art, by revolutionizing the video sector. Until recently, Text-to-Video results have been subpar and incoherent. Although the videos can be highly customized by leveraging different style and aesthetics, an overwhelming amount of existing noise made the results inconsistent.

However, OpenAI's Sora is the first video AI generative model to buck that trend. Sora is an AI model that can create realistic video clips from text prompts. It can generate videos of diverse durations, aspect ratios and resolutions. Based on the technical report by OpenAI, Sora's development team was inspired by the use of tokens (chunks of text) for LLM and used visual patches for Sora [35]. Videos are turned into patches by compressing them into a lower-dimensional latent space and then the representation is decomposed into spacetime patches. Sora is a diffusion model, in that, given noisy patches, it is trained to predict and output "clean" patches. It is also a diffusion transformer, taking cues from ChatGPT, as they found that the scaling properties of transformers can be seen carry over in video models as well.

At the time of writing of this thesis, Sora had just been announced. Therefore, the architecture and results of the model have not been properly tested by external researchers and the research has not been peer-reviewed. However, if judged by clips released by OpenAI, Sora is the first major milestone in Video Generation with astonishing results that will only improve in the future.

3.2.5 Other Models

Other than Text-to-Text, Text-to-Image and Text-to-Speech, there are other widely used models and some notable among them belong to the general Image-to-Text and Image-to-Image categories. The most common application for Image-to-Text is image captioning, the task of predicting a caption for a given image. It is usually done with encoder-decoder models or compositional methods [36].

On the other hand, the most common Image-to-Image applications are style transfer, image translation and super resolution. Style transfer refers to the process of converting a photograph into a picture of a different style.

Contrary to style transfer, image translation retains the style, but selectively alters the content. For example, a model that turns a photograph that was taken during daytime into a photograph that was taken during night time would be considered image translation.

Finally, super resolution refers to the process of enhancing an image by increasing its native resolution or sharpening its features.

3.3 Prompt Engineering

Prompt Design or Prompt Engineering is the practice of writing and refining/optimizing

textual inputs for generative systems [37]. The term was originally coined to denote the practice of writing and structuring inputs for the language model GPT-3, but has since been expanded upon and now can be used more widely as the practice of refining textual input for any generative model, whether that is Text-to-Text, Text-to-Image or any other model.

It is not a hard science, but a term that originates from within the online communities of these models. Even though it is not a hard science, its usefulness cannot be understated. The increasing complexity of the models and their nebulous architecture means that there is not always a direct correlation between the intent of the user and the resulting output of the model. Depending on the structure and length of the input prompt, two different prompts with the same meaning can yield completely different results. Therefore, the content, structure and even style of the prompt are all very important. The different techniques for prompt design differ depending on the model.

3.3.1 Prompt Engineering for Text-to-Image models

In Text-to-Image models, as Sam Witteveen et al. explain [38], any textual prompt can be divided into:

a) The factual content of the image (for example: ‘ A man holding an umbrella in the rain’), which is the main subject of the image.

b) The stylistic considerations and flourishes that dictate how the factual content is displayed.

After evaluating the Stable Diffusion model with over 2,000 prompt variations, they found that simple adjectives can have a relatively small impact on the generated image, while nouns tend to dramatically shift the image. Furthermore, using specific names of artists or styles (eg. ‘in the style of Van Gogh’) can also dramatically change the image. The prompt is usually initiated with the factual content and then, separated by commas, descriptors follow.

Internet users have also published their own guides containing certain descriptors that can be used for any prompt and can potentially elevate the result in text-to-image models, like ‘trending in artstation’ or ‘masterpiece’. A community guide for prompt engineering for Stable diffusion [39] offers several descriptors for each basic category: Subject, (the factual content), Medium, Style, Art-sharing website, Resolution, Color, Lighting, Additional Details.

3.3.2 Prompt Engineering for ChatGPT

OpenAI themselves have published an article with certain prompt engineering techniques and helpful guidelines for ChatGPT like providing reference text, writing clear instructions and asking the model to adopt a persona like an administrator or a professor [40].

After testing ChatGPT with a few prompts, it is clear that it performs better when longer prompts are broken into smaller ones, by making incremental adjustments. A similar experiment was conducted by Wei et al. of Google in 2022 [41] in what they call Chain of thought prompting, an approach that encourages LLMs to break down a complex prompt into intermediate steps and found a much boosted performance when it comes to results. As shown below, the solve rate is much higher when the user follows the chain-of-thought prompting:

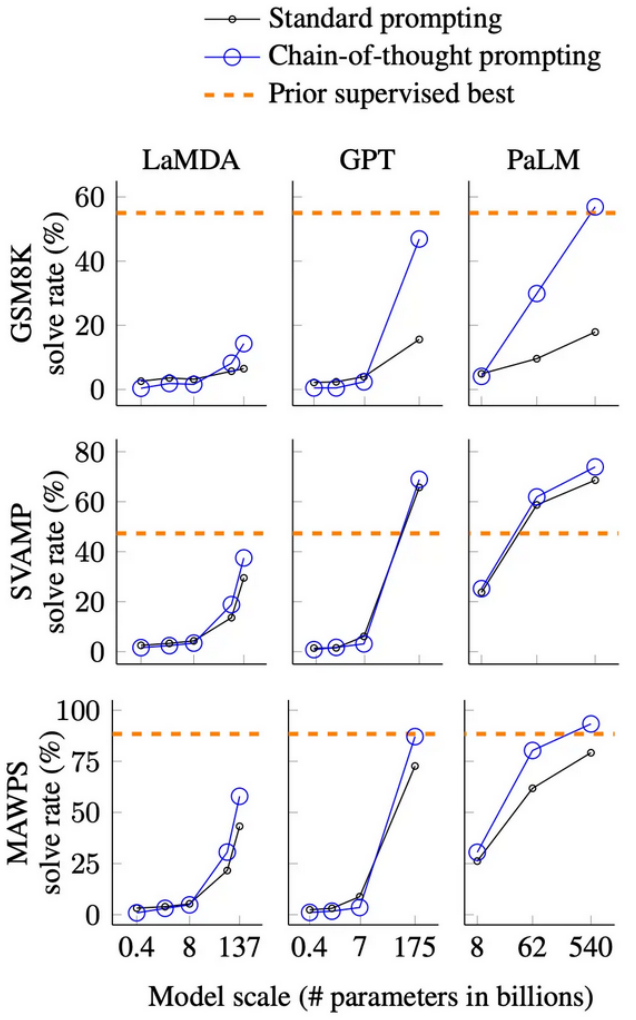

Figure 13: The performance and solve rate of chain-of-thought prompting compared to standard prompting [41].

It should be mentioned that their approach does not take into account human error. It is equally likely that part of the boosted performance is because it is easier for the user to make mistakes or be unclear if the prompt is long and consists of many sentences rather than a clear, concise one. Having said that, it is probably a combination of both.

3.4 Risks and concerns of Generative Artificial Intelligence

Having already emphasized the most popular Generative AI models and their evolution and revolutionary impact on society, it is important to dedicate this part of the thesis to bring to attention the potential concerns and dangers of the future of Generative AI, some of which have already started to appear. With the introduction of any new technology, it is the role of academic research to emphasize the risks behind its careless use.

It is easy to imagine how many things could go wrong in the future with a rapidly growing technology that emulates the human brain and while it is very specialized right now, it achieves better results than even a human would in some creative fields, and it will keep expanding. So instead of mentioning future concerns, it is more apt and productive to focus on today and the current risks that this new technology entails. The main risks and concerns can be distilled in four categories:

3.4.1 Deepfakes and imitations of voice and looks

Deepfakes are media that have been edited and manipulated to resemble someone's looks and likeness. They are a Generative Artificial Intelligence technology and many of them rely on a type of Neural Networks, called an autoencoder [42] or GANs, models with two neural networks working in tandem, a generator and a discriminator. Recent deepfake technology has been extremely accessible with dozens of apps that can be accessed via a smartphone. The technology has progressed so rapidly that it is increasingly difficult to accurately detect a deepfaked image in some cases without external, specialized tools.

This has led to more sophisticated phishing, meaning more elaborate and believable lures. Phishing is the fraudulent practice of pretending to be reputable companies or persons in order to induce individuals to reveal personal information, such as passwords and credit card numbers. For example, the persona of 'Mr. Beast', a giant Youtube and internet personality was used in a deep fake video to lure unsuspecting victims and get them to click a specific link and enter their credentials, by offering free devices [43]. The ruse counted on the fact that James Stephen Donaldson, also known as 'Mr. Beast' is known for doing charity events such as this, so it was even more believable.

AI Voice Cloning generators have also become very popular lately and are perhaps even more impressive than deepfake technology.

On a surface level, this will undoubtedly affect celebrities and famous persons with almost anyone being able to replicate their voice or looks, but it is disturbingly easy to imagine scenarios of world leaders, politicians or otherwise important figures that can influence people or institutions, being exploited to dangerous outcomes. Furthermore, this technology could be used to compromise anyone by using their likeness or voice to do or say dubious things [44]. And if everything one says and hears can be produced with very little effort using Generative AI, the value of contents shrinks dramatically [44]. As with photograph editing software in the past, when it is difficult to tell real content from edited or even generated, the value of content gets degraded.

3.4.2 Privacy, Ownership and ethical concerns

Training for these extremely big and complicated models necessitates accessing a vast amount of data, data that originates from the Internet. This has raised concerns over the Ownership and Privacy of users in two different aspects:

- When it comes to ownership, big models like ChatGPT and DALL-E have mostly been trained using the Internet as a resource. In the case of DALL-E, that means using artwork from millions of creators online without their explicit permission, the legality of which has not yet been established in court. The model produces original artwork (generative AI), but what if the artwork is highly derivative? Gowthami Somepalli et al. have researched this topic in their paper called "Diffusion Art or Digital Forgery? Investigating Data Replication in Stable Diffusion" with no conclusive results [45]. After experimenting with Stable Diffusion, they found that it reproduced very similar images, but the truth is there is no way to know what is happening inside the model and how the results are reached, which is another concern, as even the engineers of such systems do not know how the systems arrive at certain results. Strong legal precedent has not yet been set, but there have been a few trials and during a recent copyright infringement case, a US district judge in California sided with ChatGPT [46]. According to the judge, the three authors that brought the case had to substantiate not only direct correlation (that their works had in fact been used and show outputs of ChatGPT similar to their work), but intent as well and as of the writing of this thesis, they failed to do so.

• In regards to privacy, it is known that most models use data from user input to further train the models. This has resulted in Italy outright banning ChatGPT (temporarily) for having privacy concerns [47]. There have been cases of users trusting the chatbot so much to even use it as a digital therapist, but contrary to certified therapists, their data is actually saved and used for further training (in the best case scenario). Maintaining privacy and establishing trust is not only ethically imperative but also promotes broader adoption of Machine Learning technologies, as users are more likely to engage with systems they deem reliable and respectful of their privacy.

3.4.3 Hallucinations and Biased outputs

AI Hallucinations, particularly in Large Language models, are responses generated by AI that contain misleading or outright false information [48]. Furthermore, because we do not have access to the data that most models (ChatGPT included) have been trained on, we cannot know if perhaps the data is biased.

In both cases, the user might not get the full picture or might even get a false one based on data that might not even exist. As dependence on these models that act as virtual assistants grows, people tend to rely blindly on the results without cross checking and validating. Hussam Alkaissi et al. researched the potential benefits of ChatGPT in academic research, by presenting the chatbot with two medical cases and prompting it to write about these conditions and documenting the results [49]. While they found ChatGPT to be helpful in producing coherence out of bullet points and assisting in reference sorting and management, they found that the actual data it generates can be a mix of true and completely fabricated ones, concluding and proposing that full disclosure of the use of these AI tools should be placed when they are used.

3.4.4 Black box architectures - closed Datasets

While not exactly a risk or danger of the use of Generative Artificial Intelligence per say, in this section emphasis is placed on the risk of monopoly of research in these fields. Because these models and projects are easily monetizable, clearly demonstrated by the huge financial success of ChatGPT, there is incentive for secrecy and closed information systems. That means that apart from increasing demand of resources necessary to create big datasets, that even algorithms or details of models are unknown to the public. Independent research teams, including academic research teams, stand no chance when competing against big corporations in this highly financially dependent area. Even more important than highly performant algorithms is data and the number of open datasets is alarmingly low. Open datasets are publicly accessible collections of data that are made available to researchers and scientists for analysis and experimentation. They play a vital role in advancing scientific understanding and fueling innovation in various domains. By openly sharing datasets, researchers can collaborate, verify findings, and build upon existing knowledge, leading to more accurate and robust scientific discoveries. Open datasets also enable the reproducibility of research, allowing other researchers to validate and verify results, contributing to the overall reliability of scientific studies. Furthermore, open datasets foster transparency and accountability, enabling scrutiny and promoting the identification of potential biases or limitations in the data, which can ultimately lead to improved methodologies and models.

Unfortunately, most of the large datasets used to train some of the most well-known models like the very popular ChatGPT are not open. It should be the focus of the scientific community to strive towards a more open environment, particularly when it comes to data and datasets.

The closed architecture of these Generative AI systems, which is exacerbated by the

fact that most of them are not open source and the data and particularities of the algorithms and models are not public information, coupled with the nature of Deep Learning means that there is less HITL (= Human in the Loop) and past a certain point, not even the engineers of the models will be able to tell how the model reaches a decision or an output.

4. IDENTIFYING AI-GENERATED TEXT USING BPE, TF-IDF AND AN ENSEMBLE CLASSIFIER

As important and revolutionary as Generative AI is, it has raised ethical as well as practical concerns as mentioned in 3.4, especially in education. Cutting shortcuts or even cheating at workplaces and educational institutions have become commonplace, mostly because of the popularity of ChatGPT in particular. With the advancement of ChatGPT, LLMs have come surprisingly close to imitating real humans and it is often impossible to tell the difference.

The goal of this part of the thesis is, therefore, to design a model that attempts to detect and classify AI-Generated essays from real ones. The purpose of that is threefold: It is to understand how ChatGPT works and if there are underlying patterns to its results, to design and present an algorithm that is able to distinguish real from AI-Generated results and, finally, showcase more traditional techniques that perform just as well or even better than deep neural networks that have overshadowed older algorithms and techniques as new paradigms tend to do. Mostly, however, it is to show that training a customized and highly specific model on a customized dataset is not difficult and can more often than not produce much better results than using a generic AI-text detector.

The idea is inspired by the Kaggle Competition ‘LLM - Detect AI Generated Text’, which was hosted by the Vanderbilt University and the Learning Agency Lab with support by the Bill & Melinda Gates Foundation, Schmidt Futures, and the Chan Zuckerberg Initiative [50]. Before the algorithm is presented and expanded upon, emphasis is given on the current research being conducted and the State of the Art of AI-Generated text detection.

4.1 The task and State of the Art

AI-Generated text detection and classification is a fairly recent problem, as the need for solutions mostly arose with ChatGPT. In fact, it has been so influential and prominent that many educational institutions are reconsidering and reevaluating their exam structure and process.

Humans struggle to tell the difference between text generated by the latest iterations of ChatGPT and of that written by real humans, particularly if the text is meant to be formal. However, recent studies have found that there are differences and discernible patterns in AI-Generated text by ChatGPT [51][52]. Ortiz et al. found that Large Language Models (LLMs) like ChatGPT relied on a restricted vocabulary, while on a morphosyntactic level, AI-Generated texts tend to be more objective and formal. Guo et al. found more detailed and specific patterns. More specifically, ChatGPT writes in an organized matter and with long and detailed answers without spelling mistakes, harmful information and bias. It may also fabricate facts. Conversely, humans tend to lean towards subjective and informal speech with more emotion and punctuation.

In order to accurately discern if a text is AI-Generated or not and in the absence of a watermark or otherwise cooperation of the proprietor of the model, one has to rely on those patterns. At its core, the problem to detect AI-Generated text is a classification problem. A binary classification problem that relies on detecting some of the aforementioned patterns with two classes: Real and AI-Generated.

4.1.1 Methods of AI text detection

Most methods of AI text detection fall in one of the following categories: Watermarking, zero-shot detectors, fine-tuned LM Detectors, Adversarial learning methods, LLMs as Detectors and Human-assisted Methods. [53][54][55].

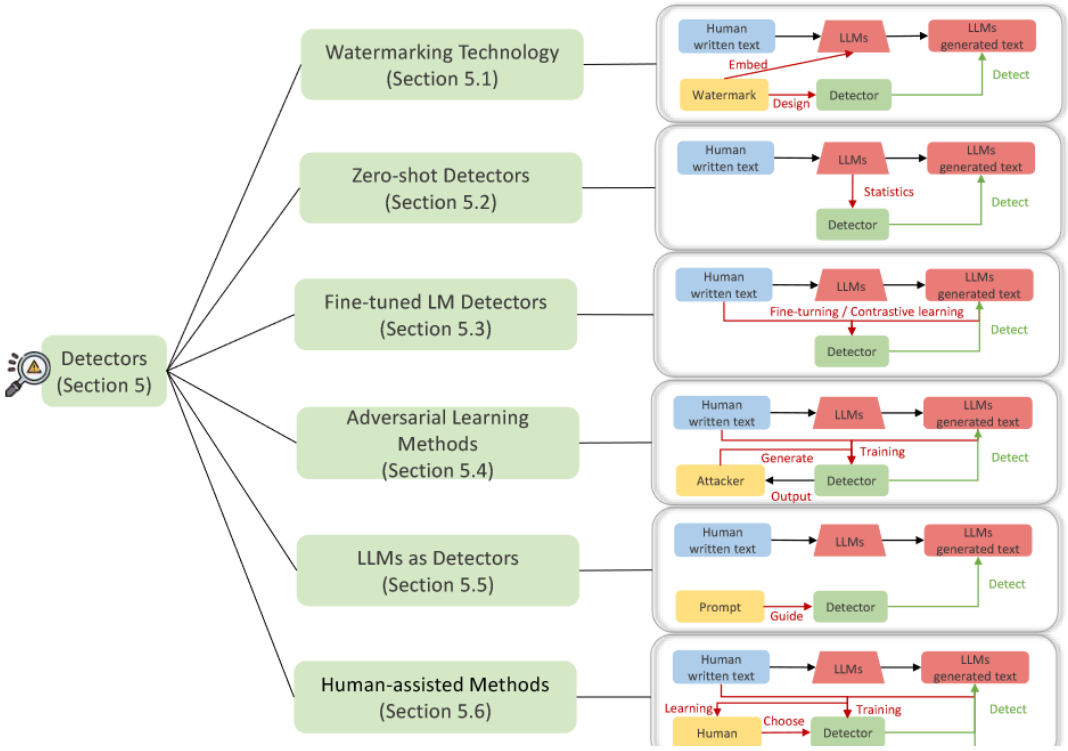

Figure 14: Different types of AI-detectors [54].

Ghosal et al. follow a more generic approach by dividing the AI-Generated text detection into two categories: Prepared and post-hoc detectors [55]. Prepared solutions involve the cooperation of the model proprietor during the text generation (as is the case with watermarking), while post-hoc detectors do not require the cooperation or the participation of the model owner and can be performed after the fact. As is obvious, these methods are much harder but more broadly applicable. Watermarking techniques, zero-shot detectors and fine-tuned LLMs or classifiers are the most common approaches to this problem.

4.1.1.1 Post-hoc methods

Two of the most prominent post-hoc methods are zero-shot detection and fine-tuning a classifier, where a dataset with real and AI-Generated text responses is needed. Contrary to that, Zero-shot detection, as its name implies, does not assume access to human-written or generated samples to detect if a text is AI-Generated or not. Instead, a probability is calculated based on specific patterns [55]. Mitchell et al. developed DetectGPT, a zero-shot detection technique, which is based on the hypothesis that samples from a model lie in areas of negative curvature of the log probability function of the model, unlike human text [56]. More specifically, if small perturbations are applied to a passage produced by the model, then the quantity $\log\left(p^{\theta}(x)\right) - \log\left(p^{\theta}(x')\right)$, where p^{θ} is the source model, x is the original passage and x' is the new passage, should be large on average if the original passage is AI-Generated compared to human text.

4.1.1.2 Prepared methods

Detection based on watermarking technology work can be defined as consisting of two

algorithms: **Watermark** and **Detect**.

The Watermark algorithm receives a language model L as input and modifies the model's outputs to include some sort of signal hidden in the generated text. The Detect algorithm takes in a text sequence *s* and a detection key *k* and outputs whether the sequence is AI-Generated or not (0 or 1 respectively). A good watermarking scheme preserves the original text and has the signal be detectable without further access to the language model.

Adversarial methods bear relevance to fine-tuning LM methods, while human-assisted methods utilize data from human knowledge and experience. That includes methods that do not directly involve humans in the active process of detecting AI text, but rather make use of collected data evaluating their prior knowledge and experience or providing humans with tools that will assist them in better distinguishing real from generated text. One such example is GLTR, a tool that applies statistical methods that improve the human detection-rate of fake text [57] or the SCARECROW framework, which facilitates the annotation review of errors produced by ChatGPT and can be a guideline/annotation system that will improve manual detection [58] [54].

While there have been interesting approaches to the AI text detection problem (like watermarking, adversarial learning, etc.), broadly speaking, the most common approaches are zero-shot detection and training a classifier.

4.1.2 Evaluation of the latest and most popular AI-text Detectors

There are many free or commercial applications and models that act as general detectors for AI-Generated text and use similar to the aforementioned techniques, the most well-known of those include OpenAI's own classifier, 'Writer', 'CROSSPLAG', 'COPYLEAKS', 'DetectGPT', 'TurnItIn' and 'GPTZERO' among others. Ahmed M. Elkhatat et al. and Weber-Wulff et al. performed comprehensive evaluations by putting the aforementioned and more models to the test and compared the results [53] [59].

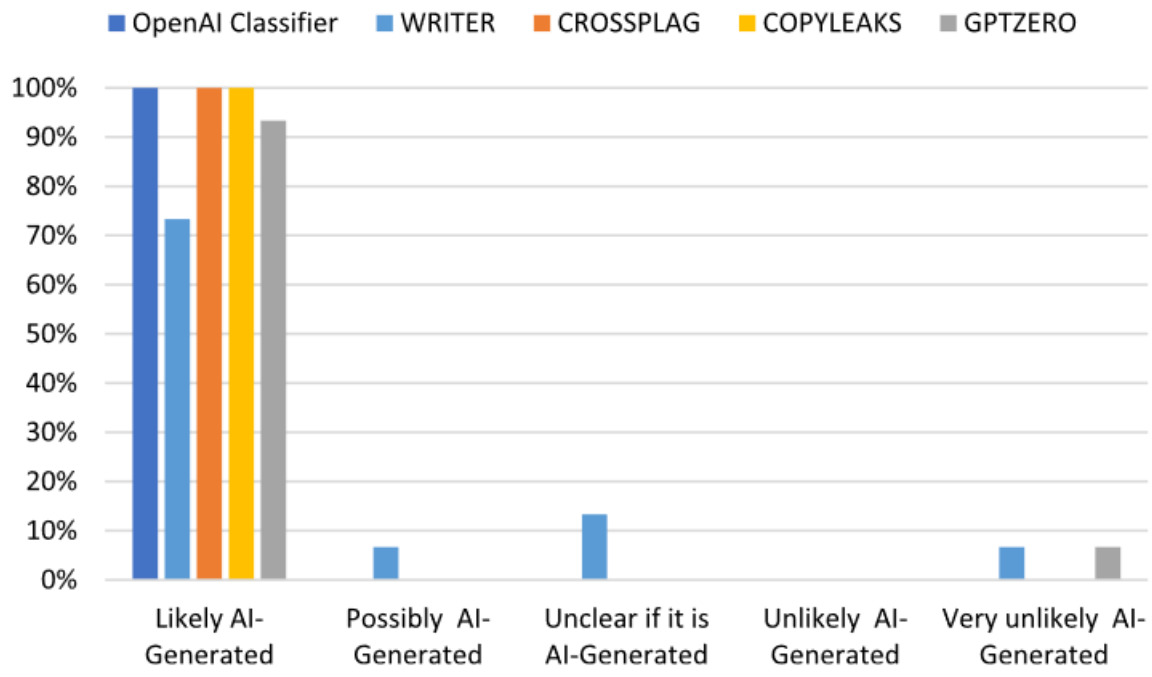

Figure 15: The responses of five AI text content detectors for GPT-3.5 generated contents [59].

Elkhatat et al. classified the diagnostic accuracy of AI detector responses into positive, negative, false positive, false negative, and uncertain. As a general outlook of the results, they found the models to be mostly consistent when it comes to text generated by ChatGPT-3.5, but very inconsistent when the more advanced ChatGPT-4 comes into play.

As shown below, the results are very impressive when it comes to ChatGPT 3.5, but fall apart if ChatGPT 4 is used to produce the outputs. It is possible that most of the models have not yet been trained using ChatGPT 4.

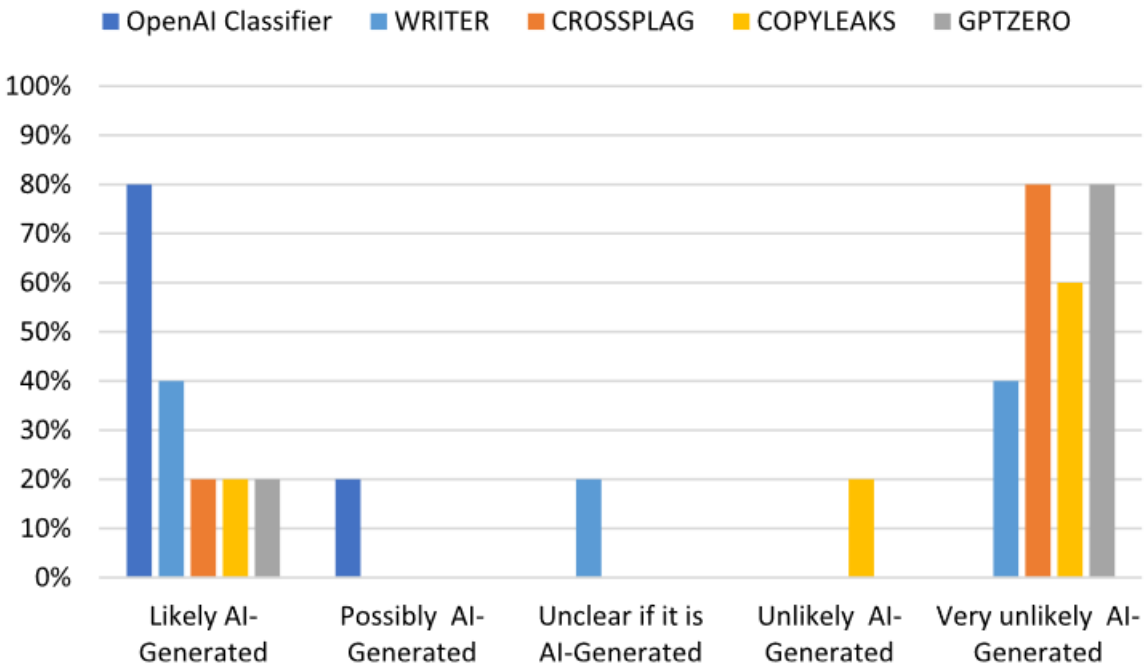

Figure 16: The responses of five AI text content detectors for GPT-4 generated contents [59].

OpenAI showed high sensitivity but low specificity, while CrossPlag showed high specificity but struggled with ChatGPT-4 even more than the others. All of the models struggled with ChatGP-4, showing perhaps a lack of adaptability when it comes to the fast evolution of AI so the authors suggest that their use is accompanied with manual review or other methods.

The results are mostly consistent with Weber-Wulff et al. 's research, which included different document types and not just human-written and AI-Generated. In their research, they included the following categories of English-language documents: human-written, human-written in a non-English language with a subsequent AI/machine translation to English, AI-Generated text, AI-Generated text with subsequent human edits and AI-Generated text with subsequent AI/machine paraphrase. Their research is comprehensive, focusing not just on average accuracy but also displaying false positives (when a text is falsely labeled as real when it is AI-Generated) and false accusations (when a text is falsely labeled as AI-Generated when it is real). Their results were slightly worse than Elkhatat et al.'s, with 'TurnItIn' and 'ZeroGPT' slightly outperforming the rest but with all scoring at an average 80% accuracy or below and only 5 of the models scoring over 70%. They found that the classification is possibly biased towards humans, as there was a clear pattern of higher accuracy when identifying human-written text. This could be intentional, because of the wide use of these tools by educators and educational institutions. Educators are believed to place a

lot of faith in AI-text detectors, which could lead to incorrect accusations due to false positives.

And that is why, particularly when it comes to education or work environments, high accuracy is arguably more important than model flexibility. At first glance, ChatGPT and similar LLMs seem to have a vast vocabulary, but when segmented into different topics, the cracks become clear and the patterns become evident. Our method uses a highly specialized Dataset and techniques to achieve much better results than general detectors and shows that training such a model is not that difficult.

4.2 The algorithm

In this part of the thesis, a post-hoc method using a fine-tuned classifier will be presented to show the incredible results that can be achieved when the dataset is very specific with a combination of real and AI-Generated essays on a few given topics. To better understand the process, the algorithm will be divided in three distinct parts:

- Data preprocessing and Tokenization using Byte-Pair Encoding.
- Using TF-IDF to extract features/vectors from essays.
- Using an ensemble-classifier to classify the essays and predict whether or not they are real or AI-Generated.

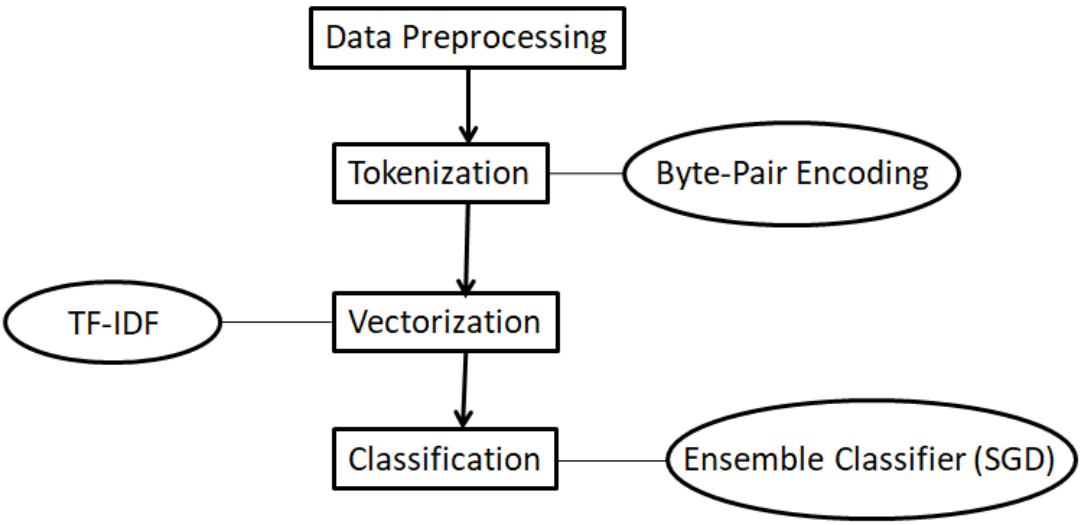

Figure 17: The algorithm – step-by-step.

4.2.1 The Dataset

The main Dataset is provided by the competition and consists of real essays written by students and AI-Generated essays (with the labels 0 and 1 respectively) [50]. All of the essays belong in one of seven different categories, as they were written in response to one of seven essay prompts. Through trial and error and by examining the contents of the dataset, the prompts are possibly the following as indicated by the community (though they have not been made public):

- ‘Exploring Venus’
- ‘The Face on Mars’

- ‘A Cowboy Who Rode the Waves’
- ‘Driverless Cars’
- ‘Facial action Coding System’
- ‘Car-free cities’
- ‘Does the electoral college work?’

In each prompt, the students were instructed to read one or more source texts and then write a response. This same information may or may not have been provided as input to an LLM when generating an essay.

Most of the competition’s essays are secret and not accessible, as they are intended for testing purposes and the structure of the dataset consists of four columns: ‘id’, ‘prompt id’, ‘text’ and ‘generated’. The ‘id’ column corresponds to a unique identifier for each row of the dataframe, while the prompt ‘id’ is a unique identifier for each of the seven prompts. ‘Text’ corresponds to the actual essay and, finally, ‘generated’ is a binary label, which indicates if the essay is AI-Generated, in which case it is 1 or 0 if it is not.

An example row of the dataset is shown below:

id	prompt_id	text	generated
0059830c	0	Cars. Cars have been around since they became famous in the 1900s, when Henry Ford created and built...	0

Figure 18: An example row of the Kaggle dataset.

The ‘id’ and ‘prompt_id’ columns are dropped, as they are not required and the ‘generated’ column is renamed to ‘label’. After enhancing and enriching the dataset with more essays (which will be mentioned in the next section of the thesis), a snapshot of the final dataset is as follows:

```
                                                    text  label
0      Cars. Cars have been around since they became ...      0
1      Transportation is a large necessity in most co...      0
2      "America's love affair with it's vehicles seem...      0
3      How often do you ride in a car? Do you drive a...      0
4      Cars are a wonderful thing. They are perhaps o...      0
...                                                  ...    ...
15866  While some find the "Face on Mars" imaged by t...      1
15867  Limiting car usage has many benefits for moder...      1
15868  The Rise of Driverless Cars\n\nThe development...      1
15869  The Open Sea Beckons\n\nThe Seagoing Cowboys p...      1
15870  While technology often progresses society in p...      1

[15871 rows x 2 columns]
```

Figure 19: An example of the final dataset.

It is important to mention that this dataset, which is comprised of real and AI-Generated essays was carefully and specifically constructed for the purposes of the competition. It was enriched by obtaining more AI-Generated essays, but most of the real essays came from the competition and many of them exhibit obvious patterns, such as grammatical errors and informal speech. As established earlier, ChatGPT generates content that rarely displays grammatical errors and is mostly formal and objective. Therefore, if there were more formal and objective real essays, the lines might have been more blurred.

4.2.2 Data Preprocessing

By researching the competition's dataset, it is evident that it is mostly comprised of real essays and not enough generated ones. Furthermore, it is quite easy to enrich the dataset with generated responses. Therefore, with the help of Kaggle users that have generated more AI responses using ChatGPT, more AI-Generated data is added to the dataset. Not unlike the original training set, the AI-Generated data is comprised of essays and results generated by issuing variations of the seven aforementioned prompts.

After enriching the dataset, the data is preprocessed. Stop words are not removed, as the performance in the results is negligible at best and worse at worst, because the goal is to detect patterns and commonly used strings, so even stop words might be helpful. However, a vital process is tokenization.

4.2.3 Byte-Pair Encoding Tokenization

Instead of a regular tokenization, Byte-Pair Encoding Tokenization is used. Byte-pair encoding is a text compression scheme introduced by Philip Gage in 1994 [60] [61]. The basic operation of the scheme is to substitute a character that did not appear in the text for a pair of two characters that appear frequently.

For example, let T be a part of the text we want to compress and

T = ABSXYABSJCSXABAB

The most frequent pair is AB, so AB is substituted for G. We then have

T1 = GSXYGSJCSXGG

Then, the most frequent pair is SX, which is substituted for F.

T2 = GFYGSJCFGG. There are no other frequent pairs, so the algorithm stops there. The original length of the text has been reduced from 16 to 10 and then a table can be encoded, which stores for every character code what it represents.

While initially developed as a compression scheme, Byte-pair Encoding is widely used in Natural Language Processing for tokenization (even used by Radford et al. for developing ChatGPT [16]). The tokenization process makes use of the Byte-Pair compression scheme, by creating a base vocabulary from the corpus and then creating merge rules by finding the most frequent pairs as in the original compression scheme.

It is an incredibly effective tokenization technique, because it can handle out of vocabulary words.

For example, in a simple tokenization into words, let us consider the sentence:

"Machine Learning is very useful."

The resulting tokenization would be 'Machine', 'Learning', 'is', 'very', 'useful'. 'Machine'

and ‘Learning’ might be unknown words so this sentence would mislead the model after training. However, in BPE, the sentence would be broken into smaller parts, for example, ‘ma’, ‘chi’, ‘ne’, etc…, that would be easier to match.

After training the tokenizing and tokenizing the dataset, a vocabulary is constructed, the size of which is determined by a parameter, which stores all of the tokens. A snapshot of the vocabulary with the vocabulary size set to 60000 is provided below:

{'Ġdr': 1387, 'ĠIdaho': 3524, 'Ġgrowing': 3243, 'Ġrepresen': 12801, 'ably': 1365, 'planet': 15390, '
aragraph': 1910, 'SE': 12314, 'Ġfost': 3468, 'ĠModel': 9112, 'Ġtechnogoly': 24050, 'ĠBobby': 25396,
'Ġlog': 4494, 'Ġexmaple': 16050, 'Ġglobally': 9280, 'Ġcombining': 16898, 'aile': 20166, 'Ġchoas': 15
001, 'Ġdidint': 30529, 'Ġfee': 9391, 'ĠPictures': 17050, 'ĠPoeple': 22447, 'erve': 2629, 'Ġthoughou
t': 17420, 'iversity': 29136, 'Ġknown': 2039, 'parents': 16604, 'OM': 26206, 'but': 3092, 'Ġbussy':
27310, 'Ġspacesh': 20787, 'iantly': 15221, 'Ġbalance': 2260, 'empl': 7401, 'Ġtear': 13965, 'Ġadjustm
ent': 18919, 'Ġglobaly': 25200, 'Ġassitance': 14163, 'ĠSecond': 4265, 'ĠUni': 4912, 'Ġessentially':
8232, 'Ġvaulable': 14471, 'Ġelectorial': 5067, 'Ġtolls': 7163, 'eting': 5231, 'Along': 9770, 'ianl':
20646, 'Ġall': 321, 'aions': 20164, 'ĠMy': 2767, 'efore': 5877, 'ride': 15233, 'Cur': 17124, 'Ġburde
n': 7188, 'Ġglitch': 6796, 'Ġcolege': 16803, 'ĠAsun': 10126, 'Ġcoqu': 27337, 'Ġhurting': 5514, 'abli
ties': 23382, 'ationships': 30461, 'Ġformat': 14310, 'Ġhating': 22725, 'vals': 26346, 'Ġnec': 17265,
'Ġcues': 11019, 'Ġcreat': 854, 'jor': 899, 'Ġona': 22792, 'ĠMyst': 24665, 'ĠMagazines': 29977, 'Ġwa
n': 5396, 'Ġordeal': 22483, 'Ġquits': 27175, 'Ġinside': 3473, 'Ġolds': 9337, 'Ġautomobilei': 27849,
'Ġscholar': 27170, 'Ġlaunches': 28291, 'Ġbiz': 13931, 'lecting': 11999, 'ĠInstitute': 7095, 'itect':
14544, 'aer': 15624, 'Ġride': 1884, 'Ġadvertisement': 16976, 'Ġevenly': 11174, 'Ġfight': 4462, 'Succ
ess': 10068, 'Ġbicicy': 26052, 'Ġwalking': 1305, 'Ġgoverner': 17446, 'Ġurgent': 15092, 'ĠINT': 17572

Figure 20: A snapshot of the vobabulary created by BPE.

4.2.4 Term Frequency – Inverse Document Frequency

After enriching the dataset given by the hosts of the Kaggle competition and tokenizing the text by utilizing the Byte-Pair Encoding tokenization, the next step is to create representations of every essay or to depict the text with a vector or a table, so that a classifier can be used to classify the vectors that represent each essay.

To vectorize each essay, TF-IDF is chosen, short for Term Frequency – Inverse Document Frequency.

TF-IDF can be defined as the calculation of how important or relevant is a word or n-gram in a corpus of documents. It is a statistical measure, an algorithm that processes all the documents in a corpus and creates a matrix, in which it stores the relevance of each word of the vocabulary for every single document.

It can be broken down to two terms: Term Frequency and Inverse Document Frequency [62].

In simple terms, term frequency is used to measure how many times a term is present in the document. So if in a document containing 1000 words, a word (‘table’ for example) is present 10 times, then

TF (table) = 10/1000 = 0.01.

As for Inverse Document Frequency, it is a weight that can be applied to Term Frequency. The inverse document frequency assigns lower weight to frequent words and greater weight for infrequent words, but the search happens across all documents. Therefore, when a word like ‘the’, which is a stop word and should be of lower consequence, appears many times, its IDF will affect the value of TF, hence TF-IDF. Let us say that our previous word, ‘table’, appears 5 times in 10 documents. Its IDF would be:

IDF (table) = $\log_e \frac{10}{5}$ = 0.3010.

To calculate the TF-IDF, we multiply TF with IDF:

TF-IDF (table) = 0.01 * 0.3010 = 0.00301.

Knowing how to calculate the TF-IDF for every word, to represent every document as a vector or a matrix, all the values for every word or phrase (unigram) in that document are calculated.

For example, this is the word relevance (TF-IDF) of the most frequent phrases of the first document in an array after performing TF-IDF on the whole corpus:

```
Top 20 features in the first document:
Feature: having carfree days, TF-IDF Score: 0.055175855951890215
Feature: has aloud, TF-IDF Score: 0.05296547731720846
Feature: having carfree, TF-IDF Score: 0.05296547731720846
Feature: it has aloud, TF-IDF Score: 0.05296547731720846
Feature: me limiting the, TF-IDF Score: 0.05296547731720846
Feature: me limiting the use, TF-IDF Score: 0.05296547731720846
Feature: to me limiting the, TF-IDF Score: 0.05296547731720846
Feature: use of cars might, TF-IDF Score: 0.05296547731720846
Feature: of cars might, TF-IDF Score: 0.05018072957383637
Feature: of cars might be, TF-IDF Score: 0.05018072957383637
Feature: says how, TF-IDF Score: 0.05009089225180768
Feature: cars might be a, TF-IDF Score: 0.04918681008305797
Feature: me limiting, TF-IDF Score: 0.04918681008305797
Feature: to me limiting, TF-IDF Score: 0.04918681008305797
Feature: be a good thing, TF-IDF Score: 0.04424137605227917
Feature: might be a good, TF-IDF Score: 0.0435093612770004
Feature: a good thing to, TF-IDF Score: 0.043053840795515406
Feature: good thing to, TF-IDF Score: 0.041821903205420924
Feature: carfree days, TF-IDF Score: 0.041629475614757
Feature: good thing to do, TF-IDF Score: 0.04140693596163204
```

Figure 21: The top 20 features in the first document, as extracted by TF-IDF.

The Byte-Pair tokenization step has been removed in that particular example, so that the resulting features of TF-IDF are more readable and understandable. As is evident from the figure, phrases are used instead of words, meaning bigrams, trigrams and quadrigrams. In fact, since the goal is to find patterns and TF-IDF does not understand ‘context’, looking exclusively for bigrams/trigrams and ignoring single words might produce even better results.

It is also important to note, that TF-IDF is an excellent but highly specialized technique, the performance of which degrades severely in large datasets. Furthermore, it uses a lot of memory, even when using batch processing in Python, meaning a function that can break data into batches to feed another function or model within a loop. Batch processing can be used to reduce the amount of memory required during training, but the matrix still needs to be stored in memory after training and it contains values for every bigram and trigram of the dataset. For example, a sentence within the dataset is: ‘to me limiting the use of cars..’ . After running TF-IDF for bigrams and trigrams, the matrix will contain values for ‘to me’, ‘me limiting’, ‘limiting the’, ‘the use’, ‘use of’, ‘of cars’, ‘to me limiting’, ‘me limiting the’, ‘limiting the use’, ‘the use of’, ‘use of cars’. When that scales into a large dataset, the model often requires obscene amounts of memory. This issue can somewhat be alleviated by using the ‘min_df’ and ‘max_df’ parameters of the Python library ‘sklearn’ version of TF-IDF.

The parameter ‘max_df’ can be used to ignore terms that have a document frequency higher than a given threshold, while ‘min_df’ can be used to ignore terms that have a document frequency lower than a given threshold. That means that ‘max_df’ can be

used to discard extremely frequent terms, while ‘min_df’ can be used to discard extremely rare terms, lowering the memory requirements, slightly. Still, TF-IDF is a very expensive algorithm that could be prohibitively expensive if the datasets are massive.

Despite being quite old, TF-IDF is one of the best methods of ‘understanding’ text, since it leaves no information behind. Other techniques were tested such as Doc2Vec, but even bag-of-words Doc2Vec is not as accurate as TF-IDF. The negative aspect of TF-IDF is mostly time and memory efficiency, but also that it misses ‘context’ or ‘nuance’, not understanding synonyms of words or the link between them. However, in this particular problem, this might be considered a boon, rather than a bane.

As previously mentioned, Byte-Pair Encoding paired with TF-IDF works particularly well, because unlike other NLP methods, TF-IDF does not understand context or nuance, but is great at creating a vast vocabulary of words/phrases and their frequency. Byte-Pair Encoding tokenization reduces the chance of finding a out of vocabulary word by breaking them down into smaller tokens, thereby increasing TF-IDF’s accuracy considerably.

4.2.5 The Classifier

Once we have the matrix from running the TF-IDF algorithm on the tokenized data, a binary classifier can be trained to classify the vectors that represent each essay and output a binary label: ‘0’ if the essay is not AI-Generated and ‘1’ if it is. Despite the documents being transformed into vectors, they still correspond to a binary label, 0 or 1, depending on them being real or AI-Generated.

4.2.5.1 Ensemble Classifier

Apart from data that follow a clearly defined relationship, it is difficult to know which classifier might be the optimal one given a specific problem and, sometimes, the heuristic approach, meaning trying different models and parameters and opting for the one that results in the best performance is the only way to proceed. To improve the accuracy of our predictions, we will use an ensemble classifier. Ensemble learning combines several machine learning models, by engaging them and combining their predictions [63].

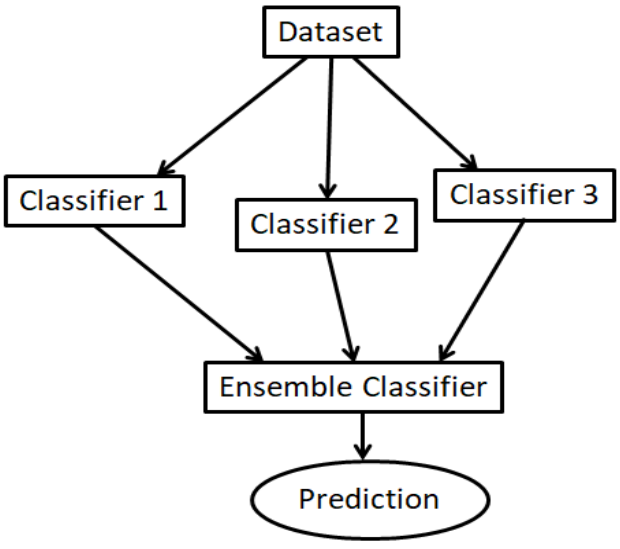

Figure 22: An ensemble classifier.

There are several different ways of combining classifiers to form an ensemble classifier. In our algorithm we use the Voting Classifier of sklearn, which uses the majority rule. In a Voting Classifier, the final prediction of the ensemble model is determined by the most common prediction. So, for example, if there are three binary classifiers and two of them predict ‘0’ as an essay’s label, then the final prediction will be ‘0’.

As for the individual models employed for the Voting Classifier, several experiments with models and combinations of models were made, while mostly trying to keep variability and variety high, so as to make the Voting Classifier more effective. The choice of classifier largely depends on the characteristics and particularities of the dataset. Therefore, a ‘free lunch’ solution is not possible. In this case, different permutations of an SGD Classifier, a Logistic Regression Classifier and a Multinomial Naïve Bayes Classifier were chosen.

4.2.5.2 SGD Classifier

SGD or Stochastic Gradient Descent is a technique that is highly efficient with linear complexity and hence it is a powerful tool for large scale learning in terms of time as compared to other machine learning methods [64].

A Stochastic Gradient Descent Classifier is a linear classifier trained and optimized by the Gradient Descent technique, which is one of the most popular algorithms for optimization and is widely used to optimize neural networks by minimizing the loss function. It is called ‘stochastic’ and is different from other gradient descent method because it employs mini-batches or random subsets in each iteration, thereby introducing an element of randomness. Contrary to that, Gradient Descent uses the entire training set to compute the gradients and not just a subset of the data.

In our model, the loss function is the modifier Huber for the SGD classifiers, but we change the max iterations of the training process.

4.2.5.3 Multinomial Naïve Bayes Classifier

A Multinomial Naïve Bayes Classifier is a supervised learning method that uses probability and is focused on text classification cases [65] [66] [67]. In general, a Naïve Bayes Classifier predicts the probability of a class given a specific token or in our case, essay. It uses the Bayes theorem to construct a Bayesian probabilistic model that assigns a posterior class probability to an instance [66]. It’s naïve because it presupposes feature independence, which means that a feature being in the dataset does not affect the presence of another feature.

The Multinomial Naïve Bayes Classifier can be considered as an upgraded version of the existing Naïve Bayes classifier, as it takes the frequency of each word into account and is, therefore, a preferred method for text classification tasks [68].

4.3 Results and Evaluation

4.3.1 The competition’s score

The competition’s evaluation metric is the AUC or Area Under the ROC Curve. A ROC Curve (receiver operating characteristic curve) is a graph showcasing the performance of a classification model, by leveraging two parameters:

a) True Positive Rate

b) False Positive Rate

The True Positive Rate can be calculated as follows:

TPR = Number of true positives / (Number of true positives + Number of False

Negatives).

The number of true positives in addition with the number of false negatives simply comprise all the AI-Generated responses. The number of AI-Generated essays that the model accurately detected and the number that were AI-Generated but the model could not detect.

Following that logic, the False Positive Rate can be calculated as follows:

FPR = Number of False Positives / (Number of False Positives + Number of True Negatives).

To form the ROC curve, the False Positive Rate or FPR is plotted on the x-axis, while the True Positive Rate or TPR is plotted on the y-axis at different thresholds. The thresholds represent the different values at which the model decides whether a predicted probability should be classified as positive or negative or as 1 or 0 respectively. So, for example, if the threshold is 0.5 and the predicted probability for an essay is greater or equal to 0.5, that is classified as an AI-Generated essay. By running the model and calculating the FPR and TPR for different thresholds, the ROC curve can be plotted.

The Area under the ROC Curve (AUC) is the entire two-dimensional area underneath the ROC curve. It is an aggregate measure of performance of the model and can be found by calculating the integral of the curve. Therefore, the AUC score of a ROC curve is a number between 0 and 1, where 1 is the maximum value when the ROC curve is the line y=1.

With just a simple structure, using specific techniques and without doing much fine-tuning or extreme preprocessing, this method scored a near **0.94** accuracy in the Kaggle competition, using a hidden test set comprised of both real and AI-Generated essays.

4.3.2 Evaluating the model outside the competition

To evaluate the model outside the competition, three sets of tests were performed. The dataset was first partitioned into training and testing subsets in a ratio of 80% to 20% respectively, following standard practice in Machine Learning experimentation. Then, another, more comprehensive test set was created by using a third-party dataset with essays that were not part of the training set. The essays were provided by Kaggle users and adhere to the guidelines of the dataset. Finally, k-fold cross validation is used on the original training set and in all instances, the results are near perfect:

Table 1: Evaluation of the model using a simple test set, an enhanced test set and using Cross Validation on the original training set.

Test set	Accuracy Score
Original Test set	0.99
Training Set (Cross Validation)	0.99
Enhanced Test Set	0.98

Furthermore, the predicted probabilities are calculated and the resulting ROC curve (on the enhanced test set) is the following:

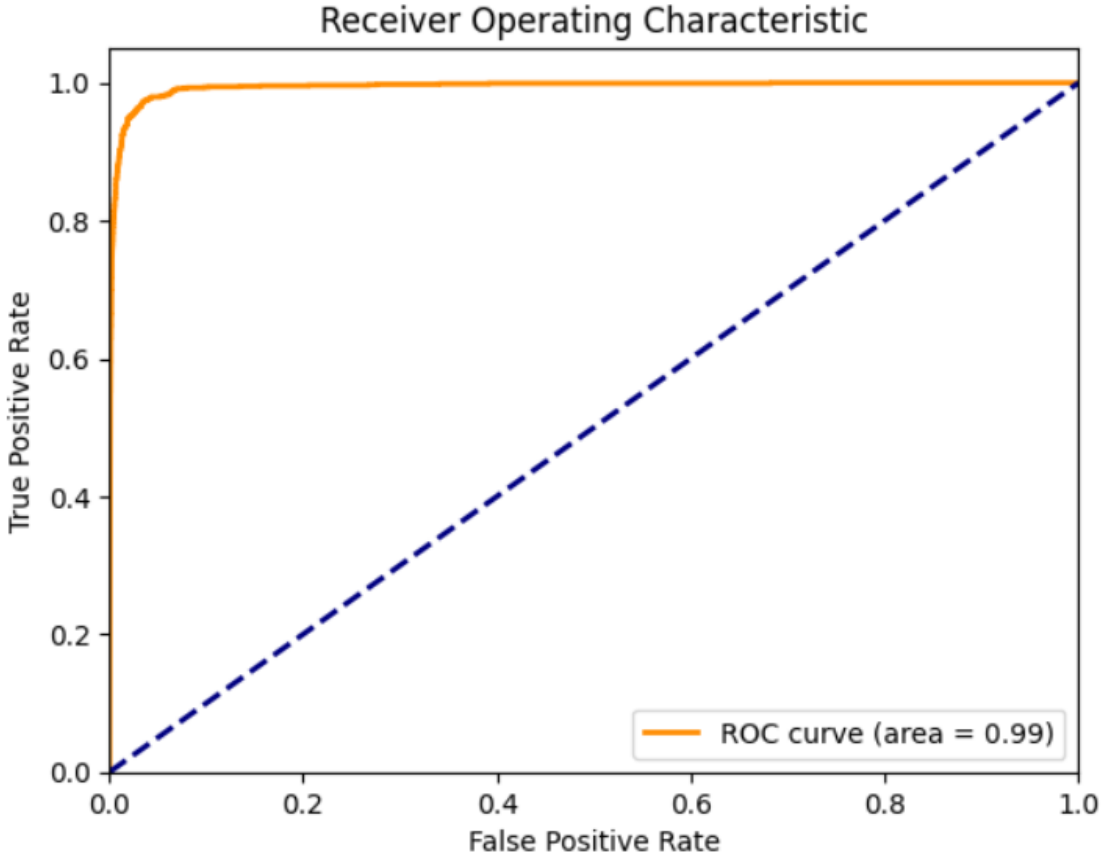

Figure 23: The AUC of the model after evaluating it on an enhanced test set.

We compare our results with the results of the popular AI detection tools by feeding them a small sample of 200 essays from the Dataset and in spite of the small size of the sample, our results are mostly consistent with Elkhatat et al. and Weber-Wulff et al.'s research. The models more accurately discern if an essay is real, but sometimes fumble the results when it comes to AI-Generated essays. Despite that, accuracy is better than the mean accuracy of those models as seen in Elkhatat et al.'s and Weber-Wulff's research, clearing 85%. We believe that is because many of the essays have obvious tells and patterns in service of the competition. Still, compared to the near perfect accuracy of our model, their results are worse, strongly indicating that when we narrow down the search area of patterns in very specific topics and we use a method like TF-IDF combined with BPE that leaves little information behind, it is easy to catch the patterns that betray if something is generated by ChatGPT.

4.4 Absence of real data and Future Work

The model produces near-perfect results, but these results hinge on a very specific and carefully constructed dataset, which was created for the competition. The two main disadvantages attributed to the method of training a customized classifier to detect AI-generated text are a) that it adheres too closely to the training set and b) how expensive it is to build a suitable dataset. Obtaining AI-Generated essays for specific topics is not expensive at all, so that raises the question of how the model will perform in two different and potentially worse real-life scenarios:

A) If there are fewer essays written by humans. It is easy to obtain AI-Generated essays, but real essays are more difficult to come by.

B) If the essays (real or AI-generated) are of a very specific topic/topics and we want to determine if an essay of a different topic is AI-Generated or not.

4.4.1 How the number of real essays within the dataset affects the accuracy of the model

To understand the impact that the number of real essays has on the results, the number of real essays is reduced, while keeping the same number of AI-Generated essays within the dataset, since they are much easier to obtain or produce. The results are displayed on the graph below, as the accuracy is plotted against the number of real essays within the dataset:

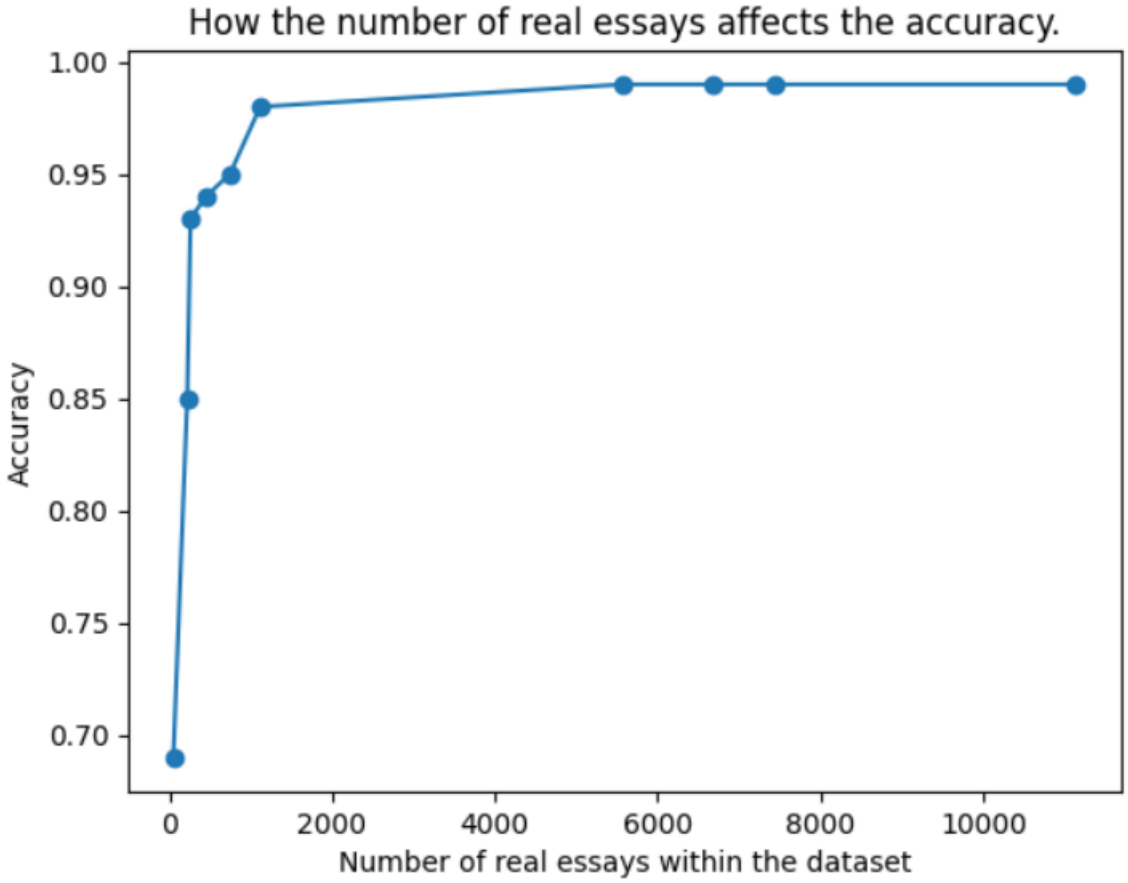

Figure 24: How the number of real essays affects the accuracy of the model.

As we can see, when the real essays are a thousand or more, even when the AI-generated ones are twice as many, the accuracy is maintained at 0.95 or even higher. In the enhanced training set, with about 11.000 real essays and 5.000 AI-Generated ones in the starting dataset, only when the number of real essays drops to 500 or fewer, does the accuracy exhibit a noticeable decrease. Therefore, even when the AI-Generated essays vastly outnumber the real essays in a 10:1 ratio, the results are quite good when we are dealing with a very specific topic and a specialized dataset, showcasing perhaps that when taken as a whole, ChatGPT and LLMs seem quite vast, but when segmented in specific topics, the limitations become obvious.

It is evident that **some** real samples are necessary, but even with just a few, the model performs reasonably well. If there are no real samples available, instead of classification we can do anomaly detection. When there is relatively balanced data of two different classes available, a model can be trained to classify them, but when there is data of just one class available, what we can do is create a hyper focused model to find the most detailed and minute common patterns, so when a new point arrives, potential deviations can be found.

4.4.2 Absence of real data within the dataset – Anomaly Detection

Suppose there are no real essays in the dataset, but unlike a zero-shot method, we do have AI-Generated essays for a specific topic that are easily obtainable. We basically want to train a model and then perform a comprehensive statistical analysis of our data, so that when a point comes (the real essays) that deviates from the other points, it is an anomaly.

There are a number of ways of tackling this problem, but we get the best results by training a one class Support Vector Machine (SVM). Support Vector Machines are a set of supervised learning methods used for classification, regression or in our case outlier detection. A one-class SVM is primarily an outlier detection tool, as it is trained on one type of data. We fine-tune the TF-IDF model to get as much information as possible and then train the one-class SVM model on the AI-Generated essays. After training, the real essays are fed into the model to obtain the predictions and the resulting accuracy is **0.93**.

We plot the test data that only contain real essays below to showcase how the model makes the decision. The line y=0 is the decision boundary and the samples above belong to the AI-Generated class, while the rest belong to the real essays. Therefore, the points above the boundary are the points the model missed or classified inaccurately.

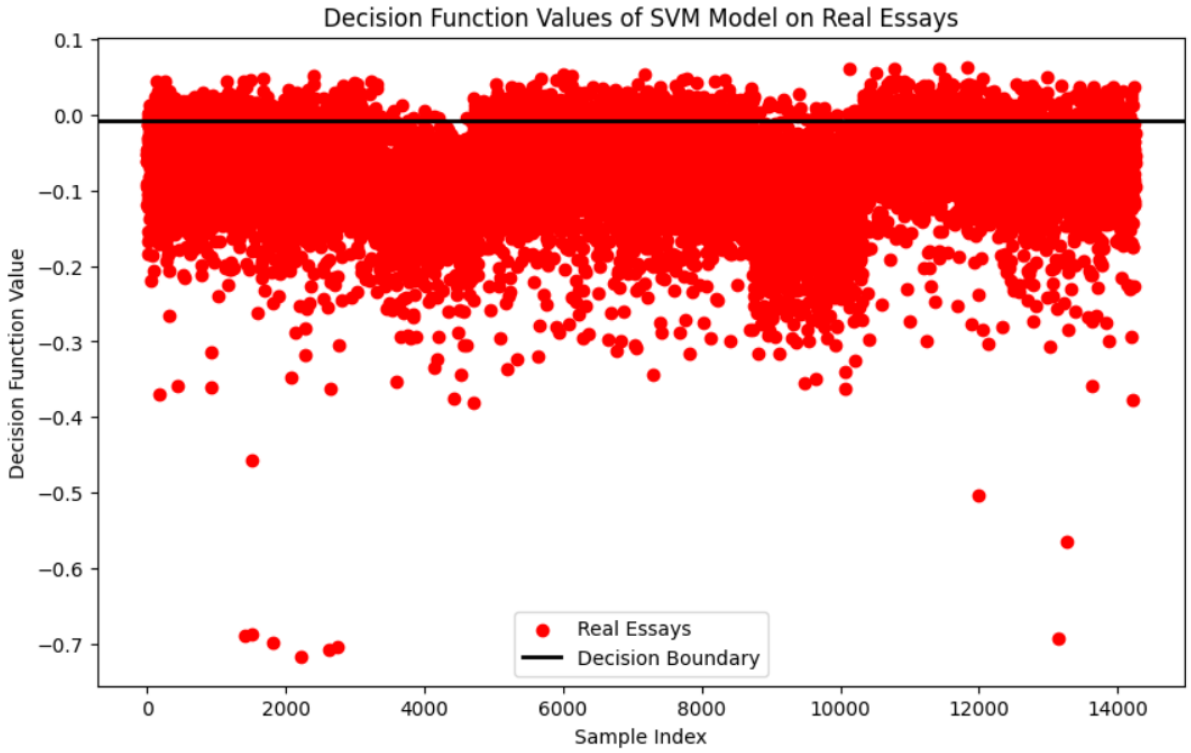

Figure 25: Decision Function Values of SVM Model on Real Essays.

In conclusion, the number of real essays is almost inconsequential, as even with just a few, good results are maintained, while if there are no real essays, a change in approach in the form of anomaly detection produces good if not great results.

On the other hand, if the test set does not adhere to the strict ‘guidelines’ of the dataset the model was trained one, in other words, if the essays we want to test are not a result of specific prompts that were used to obtain the training set, then the results can vary wildly. Using a smaller dataset of random essays, we test the model and the results are not as promising and are more often than not worse than the results of generic AI text

detectors. The techniques used in our model (such as TF-IDF) are trained to be incredibly accurate for a very specific dataset, by searching for patterns. To remedy this issue, a more generic dataset could be used with emphasis placed on stop words and common words so that more generic patterns can be obtained.

However, when it comes to AI-detectors, it is arguably better to have better accuracy, than flexibility or ease of use. In work environments or educational institutions, where accurately discerning if someone's work is AI-Generated or not could cost their job or their education, it is vital that emphasis is placed on accuracy, rather than performance or flexibility. We argue and demonstrate that training a model for specific topics works better than using generic AI-detection tools. Still, no current tool or classifier has perfect accuracy and this should be taken into account when discerning if an essay is AI-Generated has life-changing consequences. Furthermore, the dataset used for the competition is a carefully constructed dataset with obvious tells and patterns and more work and research with extensive data needs to be done for the results to be conclusive.

5. CONCLUSIONS AND FUTURE WORK

The thesis examines the unprecedented growth of Generative Artificial Intelligence and emphasis is placed on AI-Generated text detection on specialized domains. The early years of Artificial Intelligence leading up to Machine Learning are first examined, before delving into the definition of General Artificial Intelligence and what exactly it entails. Subsequently, the different models of Generative AI based on different multimedia input and output are examined and emphasis is given on the practice of prompt engineering. Lastly, some of the major risks and dangers of the use of Generative AI are presented.

In the sequel, an algorithm for accurately classifying an essay as real or AI-generated is presented along with our findings. Commercial (or free) applications have had decent success accurately determining if a text prompt is AI-Generated or not, as long as ChatGPT-3.5 is used. If ChatGPT-4 and later versions are used to generate text, the results are inconsistent at best. Therefore, a flexible and dynamic model is proposed that when trained with a dataset comprised of real and AI-Generated essays of that particular topic, can have extremely good accuracy even with ChatGPT-4. Not only that, it is demonstrated that even without real essays, a One-Class SVM model can be trained that achieves great results when the artificial data provided for training is highly specialized. In that case, it is an outlier detection problem instead of binary classification.

In conclusion, the results strongly suggest that when trying to build or use a model to detect AI-Generated text, it is better to build a dataset around a specific topic, as it becomes much easier to detect specific patterns. Furthermore, great results are maintained when the dataset contains just a few real samples and the results are good even when there are only AI-generated essays to build a one class model for anomaly detection, instead of binary classification. However, more research needs to be carried out for the results to be conclusive. Similar experiments have to be conducted for different topics and subject areas and the results have to be compared to the results of the generic AI detectors.

ABBREVIATIONS - ACRONYMS

AI	Artificial Intelligence
LLM	Large Language Model
GPT	Generative Pre-Trained Transformer
HITL	Human in the Loop
RNN	Recurrent Neural Network
GAN	Generative Adversarial Network
TF-IDF	Term Frequency – Inverse Document Frequency
BPE	Byte-Pair Encoding
SGD	Stochastic Gradient Descent
SVM	Support Vector Machine
NLP	Natural Language Processing
ROC	Receiver Operating Characteristic
AUC	Area under the ROC Curve
TPR	True Positive Rate
FPR	False Positive Rate

REFERENCES

[1] P. McCorduck (Session Chairman), Univ. of Pittsburgh! M. Minsky, MIT: 0. Selfridge , Bolt Beranek and Newman? H. A. Simon, Carnegie-Mellon University, HISTORY OF ARTIFICIAL INTELLIGENCE. https://www.ijcai.org/Proceedings/77-2/Papers/083.pdf, 1977.

[2] Haenlein, Michael & Kaplan, Andreas, A Brief History of Artificial Intelligence: On the Past, Present, and Future of Artificial Intelligence. California Management Review. 61. 000812561986492. 10.1177/0008125619864925, 2019.

[3] Rockwell Anyoha, The History of Artificial Intelligence. Harvard University, 2017.

[4] A. L. Samuel, "Some studies in machine learning using the game of checkers," in *IBM Journal of Research and Development*, vol. 44, no. 1.2, pp. 206-226, Jan. 2000, doi: 10.1147/rd.441.0206

[5] Pramila P. Shinde, "A Review of Machine Learning and Deep Learning Applications," *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*, Pune, India, 2018, pp. 1-6, doi: 10.1109/ICCUBEA.2018.8697857.

[6] Hardesty, Larry, "Explained: Neural networks". MIT News Office, 2017. Available: https://news.mit.edu/2017/explained-neural-networks-deep-learning-0414

[7] Jiachen Wan, Yang Dong, Jing-Hao Xue, Liyan Lin, Shan Du, Jia Dong, Yue Yao, Chao Li, and Hui Ma, "Polarization-based probabilistic discriminative model for quantitative characterization of cancer cells," Biomed. Opt. Express 13, 3339-3354, 2022.

[8] S. E. Haupt, J. Cowie, S. Linden, T. McCandless, B. Kosovic and S. Alessandrini, Machine Learning for Applied Weather Prediction. *IEEE 14th International Conference on e-Science (e-Science)*, Amsterdam, Netherlands, 2018, pp. 276-277, doi: 10.1109/eScience.2018.00047. keywords: {Forecasting;Wind forecasting;Machine learning;Transportation;artificial intelligence;machine learning;renewable energy;surface transportation;weather forecasting}.

[9] Yihan Cao, Siyu Li, Yixin Liu, Zhiling Yan, Yutong Day, Philip S. Yu, Lichao Sun, “A Comprehensive Survey of AI-Generated Content (AIGC): A History of Generative AI from GAN to ChatGPT”, ArXiv:2303.04226 [cs.AI], 2023.

[10] Krystal Hu, “ChatGPT sets record for fastest-growing user base - analyst note”, Reuters. Available: https://www.reuters ".com/technology/chatgpt-sets-record-fastest-growing-user-base-analyst-note-2023-02-01/

[11] Bloomberg Intelligence, “Generative AI to Become a $1.3 Trillion Market by 2032, Research Finds”, 2023. Available: https://www.bloomberg.com/company/press/generative-ai-to-become-a-1-3-trillion-market-by-2032-research-finds/

[12] Roberto Gozalo-Brizuela, Eduardo C. Garrido-Merchan, “ChatGPT is not all you need. A State of the Art Review of large Generative AI models”, arXiv:2301.04655 [cs.LG], 2023.

[13] Konstantinos I. Roumeliotis, Nikolaos D. Tselikas, “ChatGPT and Open-AI Models: A Preliminary Review”, 2023, https://doi.org/10.3390/fi15060192.

[14] T. Wu et al., "A Brief Overview of ChatGPT: The History, Status Quo and Potential Future Development," in IEEE/CAA Journal of Automatica Sinica, vol. 10, no. 5, pp. 1122-1136, May 2023, doi: 10.1109/JAS.2023.123618.

[15] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, Attention is all you need. In Advances in Neural Information Processing Systems, pages 6000–6010, 2017.

[16] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Suktskever, “Improving Language Understanding by Generative Pre-Training”, 2018.

[17] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, “Language Models are Unsupervised Multitask Learners”, 2019.

[18] Anam Nazir, Ze Wang, A Comprehensive Survey of ChatGPT: Advancements, Applications, Prospects, and Challenges, Meta-Radiology, Volume 1, Issue 2 2023 100022, ISSN 2950-1628, doi: https://doi.org/10.1016/j.metrad.2023.100022

[19] Voraprapa Nakavachara, Tanapong Potipiti, Thanee Chaiwat, “Experimenting with Generative AI: Does ChatGPT Really Increase Everyone’s Productivity?” arXiv:2403.01770 [econ.GN], 2024.

[20] Shakked Noy, Whitney Zhang, Experimental evidence on the productivity effects of generative artificial intelligence. Science 381, 187-192, 2023. DOI: 10.1126/science.adh2586

[21] Elman Mansimov, Emilio Parisotto, Jimmy Lei Ba, Ruslan Salakhutdinov, “Generating Images from Captions with Attention”, arXiv:1511.02793 [cs.LG], 2015.

[22] Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, Honglak Lee, Generative Adversarial Text to Image Synthesis, Proceedings of the 33rd International Conference on Machine Learning, PMLR 48: 1060-1069, 2016.

[23] Liu, Anting & Zhang, Shichen, Taking Text to Image for Spin via DALL-E, 2022.

[24] Chenshuang Zhang, Chaoning Zhang, Mengchun Zhang, In So Kweon, “Text-to-image Diffusion Models in Generative AI: A Survey”, arXiv:2303.07909 [cs.CV], 2023.

[25] Ziyi Change, George Alex Koulieris, Hubert P. H. Shum, “On the Design Fundamentals of Diffusion Models: A Survey”, arXiv: 2306.04542 [cs.LG], 2023.

[26] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, Jenia Jitsev, “LAION-5B: An open large-scale dataset for training next generation image-text models”, arXiv:2210.08402 [cs.CV], 2022.

[27] Andy Baio, Simon Willison, Exploring 12 Million of the 2.3 Billion Images Used to Train Stable Diffusion’s Image Generator, WAXY, 2022. Available: https://waxy.org/2022/08/exploring-12-million-of-the-images-used-to-train-stable-diffusions-image-generator/

[28] New York Times, “An A.I.-Generated Picture Won an Art Prize. Artists Aren’t Happy”, 2022. Available: https://www.nytimes.com/2022/09/02/technology/ai-artificial-intelligence-artists.html

[29] “Sony World Photography Award: Winner refuses award after revealing AI creation”, BBC, 2023. Available: https://www.bbc.com/news/entertainment-arts-65296763.

[30] Pranav Dixit, “Why Are AI-Generated Hands So Messed Up?” Buzzfeed News, 2023. Available: https://www.buzzfeednews.com/article/pranavdixit/ai-generated-art-hands-fingers-messed-up

[31] Zhang, Chenshuang & Zhang, Chaoning & Zheng, Sheng & Zhang, Mengchun & Qamar, Maryam & Bae, Sung-Ho & Kweon, In So, “A Survey on Audio Diffusion Models: Text To Speech Synthesis and Enhancement in Generative AI”, arXiv:2303.13336 [cs.SD], 2023.

[32] Myeonghun Jeong, Hyeongju Kim, Sung Jun Cheon, Byoung Jin Choi, Nam Soo Kim, “Diff-TTS: A Denoising Diffusion Model for Text-to-Speech”, arXiv:2104.01409 [eess.AS], 2021.

[33] Koichi Saito, Naoki Murata, Toshimitsu Uesaka, Chieh-Hsin Lai, Yuhta Takida, Takao Fukui, Yuki Mitsufuji, “Unsupervised vocal dereverberation with diffusion-based generative models”, arXiv:2211.04124 [eess.AS], 2022.

[34] Junhyeok Lee, Seungu Han, “NU-Wave: A Diffusion Probabilistic Model for Neural Audio Upsampling”, arXiv:2104.02321 [eess.AS], 2021.

[35] Video generation models as world simulators, OpenAI, 2024. Available: https://openai.com/research/video-generation-models-as-world-simulators.

[36] MD. Zakir Hossain, Ferdous Sohel, Mohd Fairuz Shiratuddin, and Hamid Laga, A Comprehensive Survey of Deep Learning for Image Captioning. ACM Comput. Surv. 51, 6, Article 118 (November 2019), 36 pages. https://doi.org/10.1145/3295748

[37] JONAS OPPENLAENDER, A Taxonomy of Prompt Modifiers for Text-To-Image Generation, University of Jyväskylä, Finland, 2023.

[38] Sam Witteveen, Martin Andrews, “Investigating Prompt Engineering in Diffusion Models”, arXiv: 2211.15462 [cs.CV], 2022.

[39] Stable Diffusion prompt: a definitive guide prompt guide for stable diffusion (Updated: 2024). Available: https://stable-diffusion-art.com/prompt-guide/

[40] Prompt Engineering, OpenAI, Available: https://platform.openai.com/docs/guides/prompt-engineering

[41] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, Denny Zhou, “Chain-of-Thought Prompting Elicits Reasoning in Large Language Models”, arXiv:

2201.11903 [cs.CL], 2022.

[42] Zendran M, Rusiecki A, Swapping face images with generative neural networks for deepfake technology - experimental study. In: Procedia Computer Science, pp 834–843, 2021. Doi: https://doi.org/10.1016/j.procs.2021.08.086

[43] Kahlan Rosenblatt, MrBeast calls TikTok ad showing an AI version of him a 'scam', NBC News, 2023. (Link: https://www.nbcnews.com/tech/mrbeast-ai-tiktok-ad-deepfake-rcna118596)

[44] Bendel, O. The synthetization of human voices. *AI & Soc* **34**, 83–89, 2019. https://doi.org/10.1007/s00146-017-0748-x.

[45] Gowthami Somepalli, Vasu Singla, Micah Goldblum, Jonas Geiping, Tom Goldstein; Diffusion Art or Digital Forgery? Investigating Data Replication in Diffusion Models, Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6048-6058, 2023.

[46] Judge rejects most ChatGPT copyright claims from book authors, Ashley Belanger, Ars Technica, 2024. link: https://arstechnica.com/tech-policy/2024/02/judge-sides-with-openai-dismisses-bulk-of-book-authors-copyright-claims/

[47] ChatGPT banned in Italy over privacy concerns, BBC, 2023. (link: https://www.bbc.com/news/technology-65139406).

[48] Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yolong Chen, Longyue Wang, Anh Tuan Luu, Wei Bi, Freda Shi, Shuming Shi, “Siren's Song in the AI Ocean: A Survey on Hallucination in Large Language Models”, arXiv:2309.01219[cs.CL], 2023.

[49] Alkaissi, H., & McFarlane, S. I., Artificial Hallucinations in ChatGPT: Implications in Scientific Writing. *Cureus*, *15*(2), e35179, 2023. https://doi.org/10.7759/cureus.35179.

[50] LLM – Detect AI Generated Text, The Learning Agency Lab, Vanderbilt University, Kaggle Competition, 2024. Link: https://www.kaggle.com/competitions/llm-detect-ai-generated-text.

[51] Biyang Guo, Xin Zhang, Ziyuan Wang, Minqi Jiang, Jinran Nie, Yuxuan Ding, Jianwei Yue, Yupung Wu, “How Close is ChatGPT to Human Experts? Comparison Corpus, Evaluation, and Detection”, arXiv:2301.07597 [cs.CL], 2023.

[52] Alberto Muñoz-Ortiz, Carlos Gómez-Rodríguez and David Vilares, “Contrasting Linguistic Patterns in Human and LLM-Generated Text”, arXiv:2308.09067 [cs.CL], 2023.

[53] Weber-Wulff, D., Anohina-Naumeca, A., Bjelobaba, S. *et al.* Testing of detection tools for AI-generated text. *Int J Educ Integr* **19**, 26, 2023. https://doi.org/10.1007/s40979-023-00146-z.

[54] Junchao Wu, Shu Yang, Runzhe Zhan, Yulin Yuan, Derek F. Wong Senior Member, IEEE, “A Survey on LLM-generated Text Detection: Necessity, Methods, and Future Directions.” arXiv:2310.14724 [cs.CL], 2023.

[55] Soumya Suvra Ghosal, Souradip Chakraborty, Jonas Geiping, Furong Huang, Dinesh Manocha, Amrit Singh Bedi, “Towards Possibilities & Impossibilities of AI-generated Text Detection: A Survey.” arXiv:2310.15264v1 [cs.CL], 2023.

[56] Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D. Manning, and Chelsea Finn, DetectGPT: zero-shot machine-generated text detection using probability curvature. In Proceedings of the 40th International Conference on Machine Learning (ICML'23), Vol. 202. JMLR.org, Article 1038, 24950–24962, 2023.

[57] Sebastian Gehrmann, Hendrik Strobelt, and Alexander Rush, “GLTR: Statistical Detection and Visualization of Generated Text.” In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 111–116, Florence, Italy. Association for Computational Linguistics, 2019.

[58] Yao Dou, Maxwell Forbes, Rik Koncel-Kedziorski, Noah A. Smith, and Yejin Choi, Is GPT-3 Text Indistinguishable from Human Text? Scarecrow: A Framework for Scrutinizing Machine Text. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7250–7274, Dublin, Ireland. Association for Computational Linguistics, 2022.

[59] Elkhatat, A.M., Elsaid, K. & Almeer, S. Evaluating the efficacy of AI content detection tools in differentiating between human and AI-generated text. Int J Educ Integr 19, 17 (2023). https://doi.org/10.1007/s40979-023-00140-5.

[60] Philip Gage, A New Algorithm for Data compression. 1994. Link: http://www.pennelynn.com/Documents/CUJ/HTML/94HTML/19940045.HTM

[61] Yusuke Shibata, Takuya Kida, Shuichi Fukamachi, Masayuki Takeda, Ayumi Shinohara, Takeshi Shinohara, Setsuo Arikawa, Byte pair encoding: a text compression scheme that accelerates pattern matching, 1999.

[62] Shahzad Qaiser, Ramsha Ali, Text Mining: Use of TF-IDF to Examine the Relevance of Words to Documents, International Journal of Computer Applications (0975 – 8887), Volume 181 – No.1, July 2018.

[63] Anam Yousaf, Muhammad Umer, Saima Sadiq, Saleem Ullah, Seyedali Mirjalili, Vaibhav Rupapara, Michele Nappi, "Emotion Recognition by Textual Tweets Classification Using Voting Classifier (LR-SGD)," in *IEEE Access*, vol. 9, pp. 6286-6295, 2021, doi: 10.1109/ACCESS.2020.3047831. keywords: {Social networking (online);Blogs;Sentiment analysis;Machine learning;Support vector machines;Emotion recognition;Business;Sentiment analysis;text classification;machine learning;opinion mining;emotion recognition;artificial intelligence}.

[64] D. Mittal, D. Gaurav and S. Sekhar Roy, "An effective hybridized classifier for breast cancer diagnosis", *IEEE International Conference on Advanced Intelligent Mechatronics (AIM)*, Busan, Korea (South), 2015, pp. 1026-1031, doi: 10.1109/AIM.2015.7222674.

[65] Arif Abdurrahman Farisi et al., Sentiment analysis on hotel reviews using Multinomial Naïve Bayes classifier, J. Phys.: Conf. Ser. 1192 012024, 2019.

[66] Berrar, Daniel, Bayes' Theorem and Naive Bayes Classifier. 10.1016/B978-0-12-809633-8.20473-1, 2018.

[67] P. P. Surya, L. V. Seetha and B. Subbulakshmi, "Analysis of user emotions and opinion using Multinomial Naive Bayes Classifier", 3rd International conference on Electronics, Communication and Aerospace Technology (ICECA), Coimbatore, India, 2019, pp. 410-415, doi: 10.1109/ICECA.2019.8822096. keywords: {Conferences;Python;Classification algorithms;Testing;Motion pictures;Aerospace electronics;Machine learning algorithms;Multinomial Naive Bayes;user tweets;confusion matrix;sentimental analysis;positive;negative},