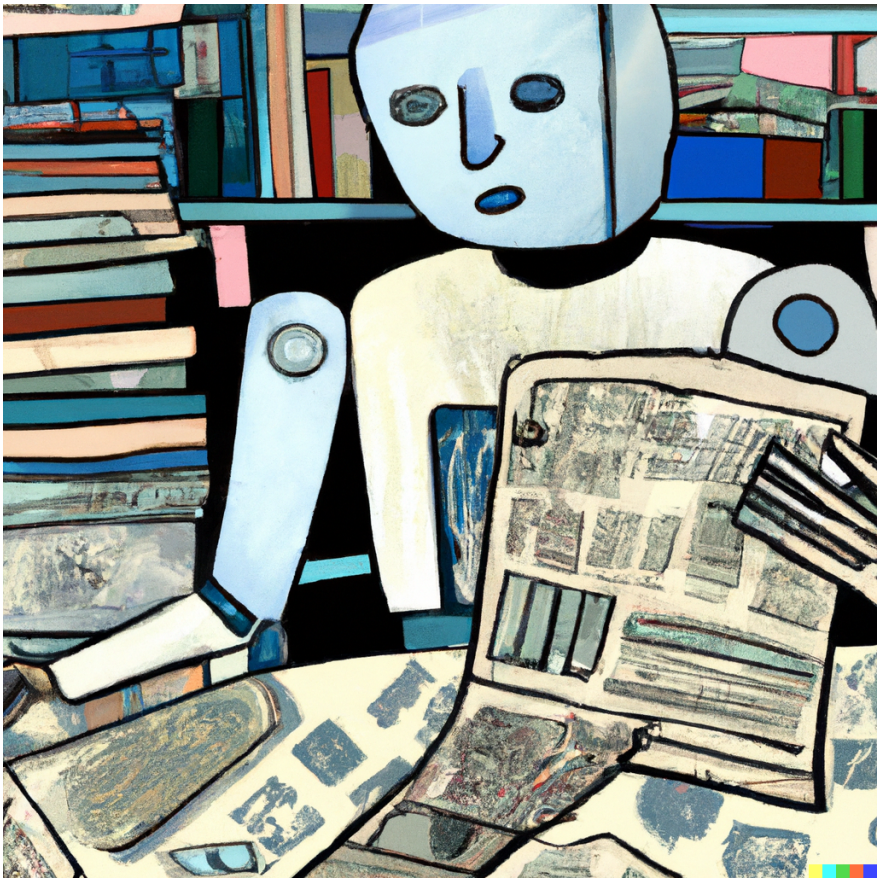

Automatic Summarization of Validated Intelligence Events

A study on GPT-3 and PRIMERA for a summarization task in a specific domain

Master's thesis in Computer science and engineering

TEO BECERRA LINUS JOHANSSON

Department of Computer Science and Engineering
CHALMERS UNIVERSITY OF TECHNOLOGY
UNIVERSITY OF GOTHENBURG
Gothenburg, Sweden 2023

Master's thesis 2023

Automatic Summarization of Validated Intelligence Events

A study on GPT-3 and PRIMERA for a summarization task in a specific domain

TEO BECERRA LINUS JOHANSSON

UNIVERSITY OF
GOTHENBURG

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Computer Science and Engineering
Chalmers University of Technology
University of Gothenburg
Gothenburg, Sweden 2023

A study on GPT-3 and PRIMERA for a summarization task in a specific domain
TEO BECERRA, LINUS JOHANSSON

Supervisor: Tobias Norlund, Department of Computer Science and Engineering
Advisor: Aron Lagerberg, Recorded Future
Examiner: Richard Johansson, Department of Computer Science and Engineering

Master's Thesis 2023
Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Cover: A robot AI writing a paper surrounded by books and newspapers in the style of Picasso. Generated by DALL-E.

Typeset in LaTeX
Gothenburg, Sweden 2023

A study on GPT-3 and PRIMERA for a summarization task in a specific domain
Teo Becerra, Linus Johansson
Department of Computer Science and Engineering
Chalmers University of Technology and University of Gothenburg

Abstract

In recent years there have been enormous progress and breakthroughs in the field of natural language processing (NLP). These breakthroughs have significantly advanced the state-of-the-art in NLP across the board and there have been a growing interest to apply these findings in an industrial setting. This thesis work is carried out in collaboration with Recorded Future, that has an interest in whether large language models can produce Validated Intelligence Event (VIE) summaries of high quality. A VIE summary is an analytical offering that describes an event in the cybersecurity domain and if these could be automatically generated it would allow for higher throughput of such summaries and would generate more value for their clients.

Our results show that such summaries are possible to produce with relatively high performance, even though they cannot be completely automated with the techniques used in this paper. However, the analysts who are currently producing these summaries expect that the use of an automated system such as this can decrease the production time by 4.

Keywords: NLP, GPT-3, PRIMERA, abstractive summarization, extractive summarization, hallucination.

Acknowledgements

We would like to give a huge thanks to our supervisors Tobias Norlund and Aron Lagerberg who has given constructive feedback, exciting views and provided us with new ideas to explore. We would also like to give a big thanks to Recorded Future, and everyone therein, for providing us the opportunity to conduct this research and providing us with their dataset and computational power. We also want to thank Richard Johansson for taking on this project to examine.

Finally, we would like to thank Chalmers University of Technology for everything these five years have had to offer. Thank you for giving us the knowledge needed to carry out this research investigation.

Teo Becerra and Linus Johansson, Gothenburg, June 2023

Contents

List of Figures

List of Tables

1

Introduction

Recent breakthroughs in the field of natural language processing (NLP), such as the introduction of large-scale language models like GPT-3 [1] and PaLM [2], have shown promising results on a wide range of applications. These applications can be found in several areas, such as in the form of chat bots (e.g. ChatGPT), in language translation systems (e.g. Google Translate) and many more. Apart from practical applications these breakthroughs have also significantly advanced the state-of-the-art in NLP across the board and have brought many new opportunities for both researchers and industry practitioners.

These large-scale language models have shifted the field of natural language processing from task-specific architectures, where one model focuses on a very specific task, to task-agnostic architectures [1]. Task-agnostic architectures are pre-trained on a huge corpus of text in an unsupervised fashion to learn a universal representation of language, which captures the underlying structure and meaning of text in a generalized manner, that can transfer to other tasks with little to no adaptation. For example [1] demonstrated in 2020 that GPT-3 performs very well in the zero-shot setting (i.e. a task it has not been specifically trained to perform) on a wide range of tasks that it was not specifically trained to perform and even reaching competitiveness with prior state-of-the-art fine-tuning approaches in the few-shot setting. The few-shot setting is where, at inference time, the model is shown a few demonstrations of the task as conditioning. However, no gradient updates are allowed in the few-shot setting.

The fact that large-scale language models can perform very well on a wide variety of downstream tasks without the need for large amounts of labeled data makes it interesting to investigate where they can be applied. One application, which is in ever-increasing demand due to the exponential growth of online content [3], is the task of automatic summarization. Large-scale learning models have already showed promising results on the task of automatic summarization such as [4] that demonstrated that GPT-3 can generate summaries comparable in quality to human written summaries and [5] that demonstrated that fine-tuning BERT can achieve state-of-the-art results on text summarization tasks.

Large language models have demonstrated impressive abilities on benchmark summarization tasks. However, to our knowledge, there has been limited research conducted on summarization tasks where there are strict requirements on the characteristics of the generated summaries. These characteristics can be that the summary should

be neutral, written for a certain audience or that it should focus on specific areas of interests in the source document(s). The aim of this thesis is to investigate how well large language models perform on summarization tasks where there are strict requirements that the generated summaries need to adhere to. The task is thus not only to compress the text but the generated summary should also adhere to a specific style guide.

This thesis is carried out in collaboration with Recorded Future, a cybersecurity company that specializes in the collection, processing, analysis and dissemination of threat intelligence. At the core of these operations are intelligence events and the production of so called Validated Intelligence Events (VIE). A VIE is daily, high quality threat intelligence, which has been verified by Recorded Future's in-house experts, that focuses on a current event that is related to cybersecurity and could for example be an ongoing cyberattack or a newly discovered vulnerability. A VIE always includes a short textual description of the event and this is what we hereafter will denote to as a VIE summary. These summaries are not simply a description of the event but should offer an informative non-biased intelligence description of the event. The task of producing these VIE summaries could potentially be automated using NLP models. This has the potential to shorten the production time of the VIE summaries and could potentially lead to higher throughput.

The utilization of large language models for such a task is not unproblematic and there are certain pitfalls. One such pitfall is hallucination where the model fabricates facts or textual content that are not present in the source document(s). This is clearly a huge problem if an automatic summarization system is to be used in an industrial setting where it is crucial that the summary is factual. One recent example of severe consequences due to a nonfactual response by a large-scale language model was the introduction of Google's conversational bot BARD. BARD generated a false fact during its demonstration which caused Google's parent company, Alphabet to lose 100$ billion in market value.

1.1 Related Work

Text summarization is a well studied problem in natural language processing, with a long history of research dating back to the early 1950s [3]. With the breakthroughs of large language models in recent years there has been a surge of interest in the research for text summarization as large language models have demonstrated state-of-the-art performance on the task of text summarization. Prior to large language models automatic summarizations systems were typically extractive and the process involved recognizing and selecting a subset of the most important sentences in the source document(s) and concatenating these to produce a summary.

In recent years the attention has shifted away from extractive summarization to abstractive summarization, which was made possible by the introduction of large language models. These summaries involve generating the summaries from scratch and go beyond simply extracting sentences from the source document(s). This is more in line with how humans would approach a summarization task and GPT-3

has even been shown to generate summaries that are preferred over human written summaries [4].

While a lot of research has been done in the field of automatic summarization, not a whole lot of the research concerns adapting the summarization task to specific domains, where there can be other domain related constraints on the summary. Most of the research on text summarization with large language models has been focused on a few open source datasets with text and summary pairs. These datasets may not be optimal to decide the capability of large language models to be fine-tuned in a more industrial setting as these datasets have a low variance (i.e. the summaries are written in the same style and the source documents are from the same website). For this reason it is interesting to see if this can be expanded to a dataset where not only are the source document(s) from a wide range of sites, but they are also of different quality.

One example of a domain specific application is the [6] report that applies text summarization to legal documents. This report compares specific summarization algorithms developed for the legal domain to general domain-independent algorithms on Indian legal documents. The domain-independent algorithms consists of both classic extractive models algorithms as well as neural network based models that can perform extractive and abstractive summarization. When it comes to quantitative ROUGE-score evaluation the classic extractive models scores high and outperforming the neural network based. However, on the qualitative human evaluation the neural network extractive model outperforms both the classic extractive and the domain specific models.

In 2019, [7] conducted a study on the summarization of microblog sites, specifically focusing on platforms like Twitter, during emergency events. The research aimed to compare various extractive summarization algorithms for this purpose. Microblogging sites play a crucial role in providing situational information during disasters or man-made emergencies. The task of summarizing microblogs inherently involves dealing with multiple documents, and is thus a multi-document summarization problem. While our own research differs in domain, it shares similarities with the study as we also utilize Twitter posts as a source and tackle a multi-document summarization problem. However, the research specifically concentrates on extractive summarization algorithms, disregarding abstractive models, and emphasizes instead the comparison of classical extractive approaches. One intriguing finding of the study is that different summarization algorithms tend to select very distinct sets of tweets for their summaries, with a low overlap. To evaluate the quality of the summaries, the summaries were compared to human-written gold standard evaluations and the overlap was scored with ROUGE scores. The domain-independent model LUHN emerged as the top performer, with the closely-following COWTS, a domain-specific model. The LUHN model identifies descriptive words and prioritizes sentences containing these words at a high frequency when constructing the summaries.

In our research we have not come across any reports on automatic summariation applied specifically in the cyber threat domain.

1.2 Aim

The purpose of this project is to asses whether it is possible to use large language models to automate or at least partially automate the production of VIE summaries. This process involves evaluating models in the zero-shot setting as well as fine-tuning models to investigate whether fine-tuning can improve the generated VIE summaries. More specifically this entails evaluating the models GPT-3 and PRIMERA in the above mentioned settings to assess whether these models can generate VIE summaries of sufficient quality.

Sufficient quality is defined as high enough quality that the VIE summaries could be published with zero or minimal human manipulation of the generated VIE summary. Firstly the thesis will investigate how the models perform in the zero-shot setting (i.e. no fine-tuning). Once the zero-shot baselines is established it will be investigated whether these models can benefit from fine-tuning in the task of generating VIE summaries.

VIE summaries typically have one to four reference documents, that are reliable references that describe the event. These references can be anything from news articles, social media posts or forums and the style of the source document(s) may therefore be vastly different compared to other open source datasets where the source document(s) are typically from the same domain (for example CNN/DM dataset).

This qualifies this investigation as a multi-document summarization task and the research questions investigated in this thesis can be expressed as:

1. *How well do current zero-shot summarization models perform on the task of generating VIE summaries?*
2. *Does fine tuning of these models improve the VIE summaries generated?*

1.3 Limitations

This thesis work will limit itself in regards to data cleaning in the sense that only text data will be investigated. Thus there will be no extraction of text from the images that may or may not be present on the website that the text data is extracted from.

Due to lack of time and resources no new language model will be trained from scratch and for the same reason only one human evaluation in collaboration with the team producing these VIE summaries will be performed.

1.4 Contributions

This project presents a human evaluation of a fine-tuned GPT-3 model for producing summaries in a specific domain. It also provides a comparison between GPT-3 and PRIMERA in both the zero-shot setting as well as the fine-tuned models.

Furthermore a second human evaluation that focuses on the sentence level hallucination is presented for GPT-3 and PRIMERA. An investigation to what extent the extractiveness of the training data impacts the extractiveness of the fine-tuned models.

1.5 Ethical Discussion

When exploring the application of large language models, such as GPT-3 or PRIMERA, for summarizing domain specific events, it is of importance to address some ethical considerations. This section highlights two key ethical aspects: the challenges in reproducing results and the potential risks associated with sharing sensitive data.

- 1. **Reproducibility Challenges**
  One of the fundamental tenets of scientific research is the ability to reproduce results and validate findings independently. However, the proprietary nature of the data which this research is based on presents a significant challenge to reproducibility. Without access to the specific training data used to fine-tune the model for summarization tasks, it becomes difficult for other researchers to replicate the results or validate the findings independently. The lack of transparency surrounding proprietary data limits the ability to assess biases, evaluate generalizability, and ensure the reliability of the summarization process. Consequently, the scientific rigor and verifiability of research outcomes can be compromised.
- 2. **Ethical Implications of Sharing Proprietary Data**
  The process of utilizing proprietary large language models often involves sending data, including potentially sensitive or proprietary information, to external platforms operated by third-party companies. This raises ethical questions regarding data privacy, security, and the potential for unauthorized access or misuse. In the context of summarizing cyber-threat related events, which may involve confidential or classified information, the transmission of such data to a foreign entity for processing introduces additional complexities and potential vulnerabilities.

2

Theory

2.1 Natural Language Processing

Natural language processing (NLP) is the study of tasks related to computational processing of natural language. These tasks can be, but are not limited to, summarization, named entity recognition (NER), sentiment analysis and machine translation. One challenge with computational processing of natural language is that language is both messy and ambiguous [8]. Furthermore, it is not obvious how text should be represented in a computational setting as words can have different meanings depending on context and two words can also have the same meaning (i.e. synonyms).

This section aims to introduce the field of natural language processing and also the models used in this thesis work.

2.1.1 Summarization

One general definition of summarization is that it is the task of compressing a piece of text into a shorter and condensed version of the original text. However, a condensed version of a text is not enough to qualify as a summary and it should also aim to capture the most crucial information and content meaning of the original text(s) [9]. The question on what the most crucial information of the original text(s) is and what the length of the summary should be are not easily defined, but there exist guidelines regarding the length and also various characteristics that can define the summaries.

The length of the summary compared to the original text(s) is typically defined by the compression rate and can be calculated by:

$$\tau = \frac{|summary|}{|source|}$$

where $|\cdot|$ is the length of the text in characters, words or sentences [9]. The compression rate should be no more than 0.5 [9] and according to [10] the best compression rate for automatic summarization systems are between 0.15 and 0.3.

The various characteristics as presented by [10] that can define a summary are outlined below.

Single document vs multi-document summarization

This is a characteristic on the input document(s) where the summary is either derived from a single input document or from many documents. In the case of multi-document summarization there is typically an expectation that the documents should be thematically related.

Domain-specific vs general

A domain-specific summary derives from input text(s) whose theme(s) are pertained to a specific domain. In this case the summary can assume less term ambiguity, idiosyncratic word and grammar usage and more which can be reflected in the summary. In the case of general summaries the original text(s) can be in any domain and the summary can therefore not make any such assumptions.

Extract vs abstract

This characteristic claims that an extract is a collection of passages (either single words or full sentences) that are extracted from the original text(s) and given verbatim as the summary. An abstract on the other hand is a newly generated text that can be produced after gaining an understanding of the original text(s).

Neutral vs evaluative

A summary can either be neutral or evaluative and applies when the original text(s) are subject to opinion or bias. A neutral summary reflects the content(s) of the original text(s) even whether it has bias or not, it does not add personal opinions or leave out any information. An evaluative summary on the other hand adds in some of the system's own bias, either explicitly (through opinion statements) or implicitly (through selective inclusion or omission of material with specific biases).

Fixed vs floating

This characteristic only applies to the summary itself and concerns the use of the summary. A fixed-situation summary is created for a specific use, a specific class of readers and situation. Thus a fixed-situation summary can conform to appropriate in-house conventions of highlighting, formatting, and so on. A floating-situation summary on the other hand can not assume any such things, but is created to be used in a variety of settings and to be read by a variety of readers.

Generic vs query-oriented

A generic summary provides the author's point of view of the original text(s) and therefore gives equal importance to all major themes in the original text(s). A query-oriented summary, sometimes called user-oriented summary, that favors specific themes or aspects in the text(s) that adheres to the users of the summary. This can be done either explicitly by highlighting certain aspects or by omitting aspects that are not deemed necessary to the users.

2.1.2 Tokenization

In order to represent text in a computational setting the text must be broken down into smaller pieces. This process is referred to as tokenization and there exists many different tokenization strategies. One such tokenization strategy is word level tokenization, which splits the text into the individual words. The individual words are then mapped to an integer representation that is kept in a vocabulary Σ to translate the word into a numerical representation. One problem with word level tokenization is that words can include declinations, conjugations or misspellings and if all these are to be included the vocabulary can easily grow into the millions. A very large vocabulary is problematic since it will require the neural network to have an enormous number of parameters. One common approach to limit the size of the vocabulary is to discard rare words and only consider a fixed amount of the most common words in the corpus. Words that are not part of the vocabulary are instead mapped to a special token representing that the word is unknown.

There is however another way to limit the vocabulary size and not force many tokens to be denoted as unknown and this is subword-based tokenization, which is a combination of word tokenization and character tokenization. This is the tokenization strategy used by most of the state-of-the-art models and ensures that the most common words are represented in the vocabulary whilst rare words are broken down into subword tokens.

Tokenization provides a way to break down a text into smaller pieces, but we want similar words to have similar representations. As every word is mapped to a single numerical value in the vocabulary the words can be represented using one-hot encoding of length V, where V is the vocabulary size. These representations could then work as the input to the network, but it turns out that there exists a more efficient way to represent the subwords. These representations are referred to as the embeddings. One way to produce this embedding is to let the neural network learn the embeddings during training. The original transformer model that is presented in Section 2.1.4 uses an embedding matrix, where all the one-hot encoded subwords are multiplied by the embedding matrix to transform the data into a vector representing the embedding. This embedding matrix is learned during training through backpropagation.

2.1.3 Large Language Models

Large language models are at the center of modern NLP and are flexible enough to adapt to various tasks without any changes. Prior to the introduction of large language models the approaches used in NLP were typically task-specific and required significant changes in architecture or pre/post-preprocessing depending on the task. Large scale language models on the other hand can easily adapt to new tasks without such major changes and have shown impressive performance on a wide range of tasks.

These large language models are based on the transformer architecture and pretrained on a huge corpora of the human language [11] in order to learn a universal

representation of the human language. This pre-training is typically done in an unsupervised fashion where the model is trained to predict the next word in a sequence. Through this pre-training the language model the model learns a mathematical model of the statistical distribution of the words in the huge corpus of human written text that it was trained on. The main benefit of this distribution is that it can sampled from, thus it is possible to provide the model with some context and thereafter sample the most probable word given this context.

Providing the model with context is typically referred to as prompting the model and does not necessarily need to be an instruction that it should follow. Assume a large language model is prompted with the question: "Who was the founder of Chalmers University of Technology?" and it responds with "William Chalmers". In reality what is being asked is what the most probable words following the given context are, not the actual answer to the question. This leads to one of the main breakthroughs of large language models, the fact that additional context can be provided to alter the statistical distribution of the most likely words given that prompt.

One way to alter the distribution and in a way teach the model to follow instructions is instruction alignment [12] where the model is taught to follow instructions. By aligning the model with the instructions the statistical distribution is altered to follow the alignment. For example, assume a large language model that has not been aligned to follow instructions is prompted with a very long text. The most likely continuation of this text may be to simply continue the text. However, if the model has previously been aligned to follow instruction and is given the instruction "Summarize text: " followed by the text in the prompt then most likely the continuation of this prompt will be a summary of the text.

Even though large language models have had much success lately there has also been limitations that have been exposed. One such example is the case of hallucination, which is when the model generates plausible but nevertheless incorrect facts.

2.1.4 Transformers

The transformer architecture was first introduced by [13] in 2017 and has since become the de-facto standard architecture for natural language processing. Prior to the introduction of the transformer architecture the dominant sequence transduction models were based on recurrent neural networks (RNNs) and convolutional neural networks (CNNs) in an encoder-decoder configuration. Typically the best of these models also connected the encoder-decoder through an attention mechanism.

The encoder-decoder configuration was introduced by [14] in 2014 and consisted of two connected recurrent neural networks. The encoder encodes a sequence of symbols into a fixed length vector representation, typically referred to as a context vector. The decoder on the other hand acts as a conditional language model, taking the context vector and the leftwards context of the target sequence to predict subsequent tokens in the target sequence. The transformer architecture builds upon the encoder-decoder model, but instead relies entirely on the attention mechanism

instead of recurrence as in [14].

A full visualization of the transformer architecture can be seen in Figure 2.1 where the left half represents the encoder and the right half represents the decoder. As previously mentioned the transformer architecture relies entirely on attention and does not involve neither convolution or recurrence, which leads to the first important difference from the original encoder-decoder configuration. This is the positional encodings that are added to the input and output embeddings and can be seen in the bottom of Figure 2.1. The reason for the introduction of positional encodings in the transformer architecture is that the attention mechanism does not have any sense of the relative or absolute position of the tokens in the sequence and therefore positional encoding is used to inject information about the positions of the tokens.

The encoders, that can be seen on the left in Figure 2.1, consist of six identical layers in the original paper that each has two sub-layers. The two sub-layers are the multi-head attention, which will be described in Section 2.1.4.1 and a fully connected feed-forward network. This fully connected feed-forward network consist of two linear transformations with a ReLU activation in-between. The input and output to the fully connected feed-forward network are of size $d_{model} = 512$ and the inner layer has dimensionality $d_{ff} = 2048$. The model also employs a residual connection [15] around both of the two sub-layers which is then followed by a layer normalization [16]. Even though the original paper places the layer normalization after the sub-layers more recent variations of the transformer architecture typically employ the layer normalization before the sub-layers [17].

The decoder is very similar to the encoder and can be seen on the right in Figure 2.1 and also consists of six identical layers in the case of the original paper. One very important difference from the encoder is that the decoder has two multi-head attention sub-layers. The masked multi-head attention sub-layer differs from the standard multi-head attention in the sense that the self-attention is masked to ensure that the prediction for output at position i only depends on the outputs at positions less than i. Another important difference is that the second multi-head attention is connected to the encoder and is used to perform multi-head attention over the output of the encoder stack.

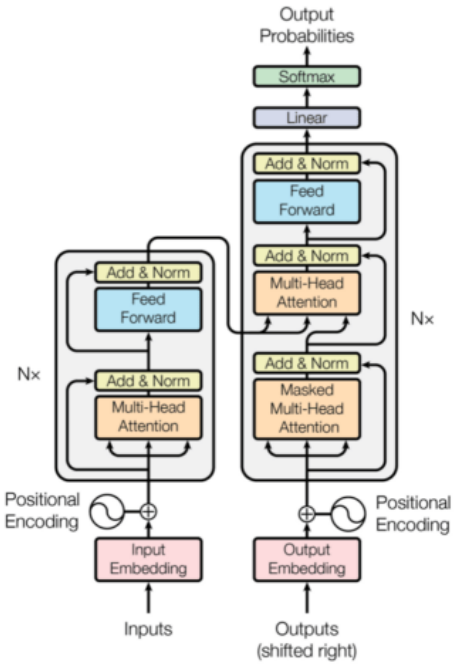

Figure 2.1: Transformer-model architecture. From "Attention Is All You Need" [1]. CC-BY.

2.1.4.1 Attention Mechanism

The attention mechanism is described in [13] as mapping a query and a set of key-value pairs to an output, where the queries, keys, values and output are all vectors. To develop an intuition for the concepts of queries and keys one can imagine a query as a token explaining what it is looking for and the key as explaining what a token is representing. Thus a token will emit a query of what it is looking for and every other token will emit a key of what they are representing. The similarity between the query and all the keys is calculated by the scaled dot product $\frac{QK^T}{\sqrt{d_k}}$ where d_k is the dimension of the keys and queries, Q is the query vector and K is a matrix of all the keys.

The output of the scaled dot product is typically called the attention score and is a measure of the similarity between a query and the keys. This attention score is then normalized using the softmax function and used to weight the contributions of the value vectors. The value vectors can be understood as a representation of what a token contributes to another token if they match. In practice the attention function can be calculated on a set of queries simultaneously by packing them in the matrix Q and also packing the keys and values into K and V. Thus the full attention function can be written as:

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

The multi-head attention mechanism is an extension of the standard self-attention

that has been described. Instead of the input only being one representation as the case in self-attention the input is transformed into multiple parallel representations. Self-attention is then calculated separately on each of these representations. The motivation behind using multi-head attention instead of the standard self-attention is that more nuanced and complex relationships can be formed between different parts of the input since it now attends to information from more representations of the sub-spaces at different positions. The individual self-attention parts that make up the multi-head attention mechanism are referred to as heads and these are concatenated to make up the final multi-head representation. In [13] they used 8 parallel attention layers or heads, but this is a hyper-parameter that varies between implementations of transformer models.

2.1.5 Longformer Encoder-Decoder Attention Mechanism

The Longformer Encoder-Decoder (LED) model, proposed by [18] in 2020 addresses the challenge of processing long sequences in traditional transformer models. The self-attention mechanism in transformers has a quadratic scaling with respect to sequence length, making it inefficient for long sequences. The LED model introduces a solution by using a sparse self-attention matrix based on an attention pattern that specifies which input locations attend to each other. This sparse self-attention mechanism allows for linear scaling with the sequence length, enabling more efficient processing of long sequences.

To allow for linear scaling with the sequence length three key modifications are made to the attention mechanism: *sliding window*, *dilated sliding window* and *global attention*.

The sliding window employs a fixed-size window attention surrounding each token and thus each token only attends to w tokens during the self-attention process, where w is the size of the window. Thus given a fixed window size of w, each token only attends to $\frac{1}{2}w$ tokens on each side. The computational complexity of this pattern is $\mathcal{O}(n \times w)$ where n is the length of the input sequence. By using many stacked layers of sliding window results in a large receptive field and the receptive field at layer l will have a receptive field of $l \times w$ if w is fixed for all layers.

Dilated sliding window is an extension to the sliding window and increases the receptive field without increasing the computation. This is performed by introducing gaps between attended tokens. Assuming a fixed window of size w for all layers and fixed gaps of size dilation d for all layers the receptive field is $l \times d \times w$.

Global attention is the third modification that is made to the attention mechanism in the LED model. This modification is needed as the dilated sliding window and sliding window are not flexible enough to learn task-specific representations. The global attention is added on a few pre-selected input locations that are chosen depending on the task. Most importantly this attention operation is symmetric, i.e. all tokens with global attention attends to all tokens in the sequence and all tokens in the sequence attend to the tokens with global attention. Since the number of tokens with global attentions is small and independent of the sequence length the

complexity of the attention mechanism is not altered by this modification.

By combining the sliding window, dilated sliding window, and global attention, the LED model achieves linear scaling with the sequence length. This linear scaling improves the efficiency of processing long sequences compared to the quadratic scaling of traditional transformers. The modifications enable the LED model to capture both local and global dependencies, effectively learn task-specific representations, and process long sequences more efficiently.

2.1.6 PRIMERA

PRIMERA is a pre-trained model for multi-document summarization that was introduced in 2021 by [9]. At the time that the PRIMERA model was introduced the advantages of pre-training and transfer learning was known for summarization tasks, but had not yet been applied to multi-document summarization tasks. Existing approaches for multi-document summarization often relied on graph-based methods that used graph neural networks to establish connections between documents, or hierarchical methods that aggregated information across intermediate representations of individual documents. However, these approaches typically required domain-specific knowledge or dataset- specific architectures, making it difficult to leverage the benefits of pre-training.

Multi-document summarization typically deals with long sequences as the source documents may be long. Therefore PRIMERA is based on a LED model, introduced in Section 2.1.5, to be able to process long sequences. The source documents are concatenated in a long sequence with a special document separator token (<doc-sep>) between the documents to mark the boundaries. This token is given global attention, which the model can use to share information across the documents.

The pre-training objective of PRIMERA is to mask out m sentences from the source documents and train the model to generate the concatenation of the m sentences as a pseudo-summary. The key idea introduced in PRIMERA is how these m sentences should be chosen such that they best represent or summarize the information in the cluster of documents. In order to identify salient sentences for masking PRIMERA starts by extracting the entities from the source documents and then select entities iteratively with the highest frequency. Sentences that include this entity are then added to a candidate set of salient sentences and the most representative sentences are selected by measuring the content overlap of the sentence w.r.t the documents other than the one it appears in.

2.1.7 GPT-3

GPT-3 is the third generation of the Generative Pretrained Transformer (GPT) model series developed by OpenAI and was introduced in 2020 by [1]. GPT-3 is based on the original transformer architecture introduced in Section 2.1.4.1 but with modifications. One major difference between the original transformer architecture is that GPT-3 does not have an encoder-decoder configuration but is decoder-only (i.e. no encoders). Further modifications to the original architecture are that the layer

normalizations were moved to the input of each sub-block and an additional layer of layer normalization was added to the final self-attention block [19]. Furthermore, GPT-3 alternates between dense and locally banded sparse attention in the layers of the transformer [1]. The specifics of the locally banded sparse attention is not known in its details, but is allegedly similar to the sparse transformer [20]. The reason why the details of the locally banded sparse attention are not known is because the GPT-3 model of families are not open sourced, but are rather owned by OpenAI.

The main difference between GPT-3 and other decoder-only transformer models is the size of the model. GPT-3 was introduced to test the hypothesis that log loss follows a smooth trend of improvement with scale. GPT-3 exists in various sizes ranging from 125 million parameters to 175 billion parameters. One difficulty of training models with that many parameters is that a very large dataset is needed. Fortunately there exists one very large dataset, which is the Common Crawl dataset that consist of nearly a trillion words. Common Crawl is a dataset that has been obtained by crawling the web and therefore the quality of data is quite low compared to a curated dataset. In order to increase the quality of the dataset the authors trained a classifier to detect low quality documents that were then removed and they also removed documents that had a high overlap with other documents. As a last curation step they also added several highly curated datasets such as English-language Wikipedia to the dataset. With this dataset the model never had to be updated on the same sequence twice, not even the biggest version of the GPT-3 with 175 billion parameters. One issue with using such a large dataset is the potential contamination on downstream tasks, which can be possible if the model has already seen the validation or test sets during pre-training.

GPT-3 has been shown to be relevant for summarization tasks in [4] and has been shown to outperform many state-of-the-art fine-tuned summarization models.

2.2 Evaluation Metrics

This section aims to describe the evaluation metrics used to evaluate the generated VIE summaries.

2.2.1 ROGUE

The ROUGE (Recall-Oriented Understudy for Gisting Evaluation) metrics were introduced by [21] in 2004 and are commonly used to evaluate the quality of automatic text summarization and machine translation systems. These metrics compare the output of such systems, typically a summary or translation, with a set of reference summaries or translations generated by humans. One of the widely used components of the ROUGE metrics is ROUGE-N, which focuses on comparing n-grams between the candidate and reference summaries.

ROUGE-N assesses the effectiveness of text summarization by measuring the overlap of n-grams between a candidate summary and a reference summary. The "N" in ROUGE-N represents the length of the n-grams used for comparison. For example,

ROUGE-1 compares the overlap of unigrams (individual words), while ROUGE-2 evaluates the agreement of bigrams (pairs of words).

To compute ROUGE-N, precision and recall values are calculated. The precision represents the ratio of the number of matching n-grams between the candidate and reference summaries to the total number of n-grams in the candidate summary. Similarly, the recall is determined by dividing the number of matching n-grams by the total number of n-grams in the reference summary. The F1 score, which is the harmonic mean of precision and recall, can also be calculated to provide an overall measure of similarity between the candidate and reference summaries.

The formulas for ROUGE-N are as follows:

$$\text{ROUGE-N}_{precision} = \frac{\text{Number of matching n-grams between candidate and reference}}{\text{Total number of n-grams in candidate summary}}$$

$$\text{ROUGE-N}_{recall} = \frac{\text{Number of matching n-grams between candidate and reference}}{\text{Total number of n-grams in reference summary}}$$

$$\text{ROUGE-N}_{F1} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Another commonly used metric from the ROUGE family is the ROUGE-L metric. ROUGE-L measures the longest common subsequences (LCS) between a candidate summary and a set of reference summaries. It quantifies the similarity by considering the union of the longest subsequences between the candidate and each reference summary. Mathematically, the union of the longest subsequences is defined as:

$$\text{LCS}_{\cup}(\text{candidate, reference}) = \cup_{r_i \in \text{reference}} \{w | w \in \text{LCS}(\text{candidate}, \text{r}_\text{i})\}$$

Here, $\text{LCS}(\text{candidate}, \text{r}_\text{i})$ represents the set of longest common subsequences between the candidate summary and the reference summary r_i.

To calculate the ROUGE-L score, precision and recall values are required. The precision score, denoted as $P_{\text{lcs}}(\text{candidate, reference})$, is the ratio of the total number of longest common subsequences between the candidate and reference summaries to the total number of words in the candidate summary. Similarly, the recall score, denoted as $R_{\text{lcs}}(\text{candidate, reference})$, is the ratio of the total number of longest common subsequences to the total number of words in the reference summary.

The ROUGE-L metric is then computed as an F1 score, which balances precision and recall, using the formula:

$$\text{ROUGE-L}(\text{candidate, reference}) = \frac{(1+\beta^2) R_{\text{lcs}}(\text{candidate, reference}) P_{\text{lcs}}(\text{candidate, reference})}{R_{\text{lcs}}(\text{candidate, reference}) + \beta^2 P_{\text{lcs}}(\text{candidate, reference})}$$

In this formula, the parameter β determines the relative importance of precision and recall. Typically, a high value of β is chosen as ROUGE scores tend to favor recall.

2.2.2 BLEU

The BLEU metric is a popular method for evaluating the quality of text summarization models, and was introduced in 2002 by researchers at IBM [22]. BLEU stands for bilingual evaluation understudy, and it was originally developed for machine translation, but it has since been adapted for variety of tasks within NLP-studies such as text summarization.

The BLEU metric works by comparing the output of a machine-generated text with one or more reference texts and calculates the degree of overlap between n-grams (sequences of n words) in the machine-generated text and the reference text. The degree of overlap is then used to compute a score between 0 and 1, where 1 indicates perfect overlap between the machine-generated text and the reference text.

BLEU score is calculated using modified precision between a candidate and a reference sentence. Given a candidate sentence $\hat{y}$ and a reference sentence y precision is calculated using the following:

$$p_n(\{\hat{y}\};\{y\}) = \frac{\sum_{s\in G_n(\hat{y})} \min(C(s,\hat{y}), C(s,y))}{\sum_{s\in G_n(\hat{y})} C(s,\hat{y})}$$

Here G_n denotes the set of n-grams and $C(s,y)$ denotes the count of the appearance of substring s in string y. For a string $y = y_1 y_2 \cdots y_K$ and any integer $n \geq 1$, the set of n-grams of y is defined as

$$G_n(y) = \{y_1 \cdots y_n, y_2 \cdots y_{n+1}, \cdots, y_{K-n+1} \cdots y_K\}$$

It is important to note that $G_n(y)$ is a set of unique elements, not a multiset that allows redundant elements. For instance, $G_2(abab) = ab, ba$.

Moreover, for two strings s and y, the substring count $C(s,y)$ is defined as the number of appearances of s as a substring of y. For example, $C(ab, abcba) = 2$.

To avoid giving high scores to sentences missing pivotal information BLEU score calculates a penalty called brevity penalty. The brevity penalty is the correction factor that is applied to the BLEU score to account for the fact that shorter summaries are more likely to achieve a high BLEU score by chance. The brevity penalty is calculated as the ratio of the length of the candidate summary to the length of the reference summary.

$$BP(\hat{S};S) := e^{-(r/c-1)^+}$$

Here c refers to the length of the candidate summary, r the length of the reference summary and $(r/c-1)^+ = \max(0, r/c-1)$, the positive part of $r/c-1$. If the candidate summary is longer than the reference summary, the brevity penalty is set to 1 as not to punish long candidates. In other cases the brevity penalty $BP = e^{1-r/c}$

$$\text{BP} = \begin{cases} 1 & \text{if } r < c \\ e^{(1-r/c)} & \text{if } c \leq r \end{cases}$$

BLEU does not have a single definition but rather a family of definitions, parameterized by the weighting vector $w = (w_1, w_2, \cdots)$. The weighting vector is a probability distribution on $1, 2, 3, \cdots$, which means that $\sum_{i=1}^{\infty} w_i = 1$, and for any $i \in 1, 2, 3, \cdots$, $w_i \in [0, 1]$.

For a given w, the BLEU score is calculated as:

$$BLEU_w(\hat{S}; S) := BP(\hat{S}; S) \cdot \exp\left(\sum_{n=1}^{\infty} w_n \ln p_n(\hat{S}; S)\right)$$

where $BP(\hat{S}; S)$ is the brevity penalty and $p_n(\hat{S}; S)$ is the modified n-gram precision.

In essence, BLEU score is a weighted geometric mean of all the modified n-gram precisions, multiplied by the brevity penalty. The most common weighting vector is $w_1 = w_2 = w_3 = w_4 = \frac{1}{4}$, as recommended in the original paper [22].

One advantage of BLEU is that it is relatively easy (and thus affordable) to compute, and it provides a single score that can be used to compare different models or variations of the same model. However, it is not without its limitations. For example since the metric only considers n-gram overlap more subtle aspects of summarization quality, such as coherence or fluency are lost.

2.2.3 BERTScore

BERTScore [23] was introduced in 2019 and addresses the common pitfalls that n-gram based metrics such as ROUGE and BLEU suffer. Specifically BERTScore addresses the issue where semantically correct phrases are penalized because they differ on the surface from the reference summary. Looking at an example from the BERTScore research paper we can highlight some of the pitfalls of the n-gram metrics: Given the reference sentence "People like foreign cars", n-gram metrics (such as BLEU) incorrectly give a higher score to the sentence "people like visiting places abroad" compared to "consumers prefer imported cars".

The BERT model uses contextual embeddings to encode both the candidate and reference texts into numerical representations, which are then used to calculate the similarity score. Using contextual embedding BERTScore can generate different vector representations of the same word given depending on the context.

Contextual embeddings aim to represent words or phrases in a way that captures their meaning and context. Where prior traditional word embeddings represent words as fixed vectors based on their frequency or co-occurrence with other words in a corpus, contextual embeddings capture the specific use of a token in a sentence, take the surrounding words into account and the overall context of the sentence to capture sequence information. Unlike word embeddings contextual embeddings can generate different vector representations for the same word in different sentences, depending on surrounding words, which form the context for the target word.

Given a reference summary $x = (x_1, ..., x_k)$ and a candidate summary $\hat{x} = (\hat{x_1}, ..., \hat{x_l})$ BERTScore uses the BERT model [24] to tokenize the text into a sequence of word

pieces. Then a representation vector for each word piece is computed with a transformer encoder, by repeatedly applying self-attention and nonlinear transformations.

Vector representation allows for soft measure of similarity instead of exact-string matching. The cosine similarity between the vector representations of a reference token x_i and a candidate token $\hat{x_j}$ is calculated by the following formula:

$$\frac{x_i^\top \hat{x_j}}{||x_i||\,||\hat{x_j}||}$$

Using pre-normalized vectors, the calculation is reduced to:

$$x_i^\top \hat{x_j}$$

Because of the contextual embedding, each vector representation of a token contain information from the rest of the sentence.

To compute the complete BERTScore each token in x is matched to a token in $\hat{x}$ to compute recall and each token in $\hat{x}$ to x to compute precision. BERTScore uses greedy matching to maximize the matching similarity score, each token is matched to the most similiar token in the other sentence. Precision and recall are then combined to compute the F1 measure.

$$R_{\text{BERT}} = \frac{1}{|x|}\sum_{x_i \in x} \max_{\hat{x}_j \in \hat{x}} \mathbf{x}_i^\top \hat{\mathbf{x}}_j, \quad P_{\text{BERT}} = \frac{1}{|\hat{x}|}\sum_{\hat{x}_j \in \hat{x}} \max_{x_i \in x} \mathbf{x}_i^\top \hat{\mathbf{x}}_j, \quad F_{\text{BERT}} = 2\frac{P_{\text{BERT}} \cdot R_{\text{BERT}}}{P_{\text{BERT}} + R_{\text{BERT}}}$$

2.2.4 Human Evaluation

Human evaluation is an important aspect when evaluating the effectiveness of automated text generation and often considered the gold standard [25]. While automated metrics, such as BLEU and ROUGE scores, can provide a quantitative measure of the quality of generated text, human evaluation is often necessary to obtain a more comprehensive understanding and determine if the generated text is acceptable for the intended audience. Thus human evaluation should, if possible, always be used when measuring the quality of automatically generated text while automatic evaluation can be used as a complement. However, human evaluation is both costly and time consuming so it is often not possible to include human evaluation in evaluation of text generation. Another challenge with human evaluation is the lack of standardization and reproducible evaluations. Even though automatic text generation has a rich history, evaluation methods has varied especially in the way that human evaluation is carried out [26].

When designing a human evaluation study, it is important to determine the appropriate evaluation criteria. Criteria should be relevant to the specific application and be defined clearly and unambiguously. Some common and generic evaluation

criteria include fluency, coherence, relevance, grammatical, and overall quality. Furthermore, specific criteria tailored to the application of the text is often included to determine its performance regarding to specific applications and intended audience.

Another important consideration when designing a human evaluation study is the choice of evaluators. Evaluators should be representative of the intended audience for the generated text. For example, if the generated text is intended for medical professionals, then the evaluators should have a background in medicine or a related field. Furthermore, evaluators may have different preferences and biases, which can affect their evaluation. Thus representation should be taken into consideration when choosing evaluators.

3

Methods

This section describes the methods use for evaluating the performance of PRIMERA [27] and GPT-3 [1] in generating unique VIE summaries within the cybersecurity domain. The main objective is to investigate whether fine-tuning these models can enhance the quality of the generated summaries. To address this objective, the following research questions are considered:

1. How well do PRIMERA and GPT-3 perform in the task of generating domain-specific summaries in the zero-shot setting?
2. Does fine-tuning of these models improve the quality of the generated domain-specific summaries?

This chapter is divided into a section concerning the datasets used and the corresponding data preprocessing, a section concerning the models and the setup used to produce zero-shot summaries as well as fine-tuning. Furthermore a section about the evaluation and the generation strategy is included.

3.1 The VIE Task

Validated intelligence events are daily, high-quality threat intelligence provided by the Insikt Group, Recorded Future's threat research division, that are designed to help clients reduce their risk. These events frequently involve cybersecurity related incidents such as data leaks and cyberattacks. The production of these VIE:s consists of gathering information and selecting stories that are deemed to provide value to the clients. Once a story has been selected and trustworthy sources have been collected a *note* that summarizes the VIE is written by the analysts and then published on the Recorded Future platform. This note is denoted in this paper as a VIE summary as it very closely resembles a summary. However, the aim of the VIE summary is not only to provide a pure summary but should rather be an analytical offering that provides insights to the clients. The VIE summaries are written with a specific rule set detailing the voice (informative, non-biased) and what content is to be included in the summary (what, who, where, when and why). This provides a significant increase in difficulty compared to a regular summarization task which may only need to satisfy basic requirements such as being informative and fluent.

The Insikt Group is Recorded Futures threat research division, consisting of analysts and security researchers with deep government, law enforcement, military, and

intelligence agency experience. Their mission is to produce intelligence on a range of cyber- and geopolitical threats that reduces risk for clients and prevents business disruption. Coverage areas encompass research domains such as state-sponsored threat groups; financially-motivated threat actors on the darknet and criminal underground; newly emerging malware and attacker infrastructure; strategic geopolitics; and influence operations. One such part of the produced information is Insikt Notes and Validated intelligence events (VIE).

Table 3.1 presents an example of a VIE summaries. Here it can be seen that the style is very straight to the point and very focused on what has happened, why it happened and also providing detailed information about the various actors involved. One main difference here from a pure summary is that a VIE summary should always aim to include all the actors that were either involved or affected by the cybersecurity event described in the VIE. Furthermore, it should also describe these actors properly if the VIE summary is written according to the writing specifications. Actors are the most important information as it provides insight to Recorded Future's clients whether a service they are using or a company that they are partnering with is affected by the incident.

Table 3.1: Examples of VIE summaries.

Example 1
Barcelona-based online survey and form building service Typeform announced a data breach today after an unknown attacker downloaded a backup file containing sensitive customer information. The backup file contained data gathered by Typeform customers through surveys and online forms up until May 3, 2018. Passwords and user payment card information was not included in the backup file the attacker took from Typeform's servers. The incident happened after the attacker exploited a vulnerability, yet the company did not reveal what vulnerability that was. Typeform did say they plugged the security hole. Typeform said its employees became aware of the breach on Wednesday, June 27, at 14:00 CET, and secured the affected server 30 minutes later.

3.2 Dataset

The dataset used in this thesis was provided by the in-house experts at Recorded Future and consisted of 2729 VIEs. Every data point in the dataset consisted of a unique ID, a VIE summary, a publication date, URLs to the source document(s) that were used in the production of the VIE summary and the entities present in the VIE summary.

3.2.1 Entities

In the context of natural language processing an entity refers to a specific, named object that can be recognized and classified in a text or sentence. Typical examples of named entities are names and locations. The in-house tool used by Recorded Future to classify entities classifies the typical entities such as names and locations, but goes a few step further and also classifies entities such as malware, attack vectors and vulnerabilities.

Therefore the entities that are extracted by the in-house tool used in this thesis to extract entities includes entities that may not be typical in contexts outside the cybersecurity domain.

3.2.2 Data Cleaning

One limitation that was imposed on the dataset prior to the data cleaning was to only include validated intelligence events where the number of sentences in the VIE summary was between three and six. Even though the VIE summaries in the dataset was of high quality it was found to require substantial cleaning. This was due to the fact that the source document(s) were referenced by URLs to the websites (typically news articles). The issue that arose was that not all URLs were readily available and that they could refer to any website that the analyst had utilized as a reference during the production of the VIE summary. Consequently, certain websites could be unavailable and the parsing was not trivial as proper parsing for every website would require a distinct set of rules for each particular website.

In light of the aforementioned issues, it was imperative to take corrective measures to address these concerns. Given the use of the dataset for a summarization task, it was essential to ensure that all the information conveyed in the VIE summary could also be located in the source document(s). This was essential since if the models did not have access to the same information as the analyst who originally authored the VIE summary it would have been no way to accurately compare these summaries during the evaluation. Consequently, the dataset was subjected to a rigorous cleaning process involving the steps outlined below.

Unavailable URLs

The data points could have one or more source documents referenced during the production of the VIE summary. If any of these source documents were unavailable the data point was discarded from the dataset.

Non-English sources

As PRIMERA [27] was one of the models investigated, which has only been pre-trained on English sources, all data points that had a reference to a non-English source were removed.

Fuzzy matching on entities

This constraint on the dataset was added to ensure that if the source document(s) were not properly parsed, then at least they would contain the most important in-

formation. The most important information was regarded as the entities that were present in the VIE summary. These entities were identified using an in-house tool provided by Recorded Future, which not only extracted named entities, but also identified other crucial terms, such as the type of attack in the case of a cybersecurity breach. The presence of these entities in the source document(s) was deemed essential, as if they were not present it would be impossible to generate a VIE summary that concerned the same information as the reference VIE summary.

In order to eliminate data points where all the entities present in the VIE summary were not present in the source document(s), the use of fuzzy matching was employed. The entity did not have to be present in each of the source documents, but it was enough that it was included in at least one. Since an entity did not necessarily have to be a single word all n-grams were considered in the source document(s), where n represents the number of words in the entity. The Levenshtein distance, which is a metric for measuring the difference between two strings, was then calculated between all such n-grams and the entity. A suitable threshold was established which seemed to most accurately correlate with a match, this was done heuristically. In cases where no n-grams had a Levenshtein distance above the certain threshold the corresponding data point was removed from the dataset.

Short source documents

If the total length of the source document(s) were less than 200 characters (around 50 tokens for GPT-3) the data point was dropped.

Token length

The base version of GPT-3 which supports fine-tuning only allowed for 2048 tokens for both the prompt and completion. Therefore every data point that would require a prompt of more than 1748 tokens were discarded. 1748 tokens were chosen as a VIE summary was typically between 200-300 tokens in length.

Manual curation

The last step consisted of a brief reading of all the fetched and parsed source documents to ensure that there was no major flaw with them.

The final processed and clean dataset consisted of 1438 validated intelligence events and was split into 1164 (80%) training data, 146 (10%) validation data and 146 (10%) test data.

3.2.3 Abstractive Dataset

One characteristic of a summary is that a summary can either be an *extract* or an *abstract*. The difference between these two are presented in Section 2.1.1 and concern whether the summary is an extract of collection of passages or if it is newly generated text. However, it is also possible for a summary to be a combination of an extract and an abstract in the case that some sentences are extracted from the source document(s) while others are not. It could also be the case that some sentences were partly extracted (i.e. only a few words differ in the sentences).

VIE summaries should not contain extracts from the source document(s) and in the cases where there were extracts from the source document(s) the Insikt group considered this as plagiarism. To determine the level of plagiarism, which we denote as extractiveness, we measure how many of the sentences that are extracts from the source document(s). More specifically we measured the following types of extractiveness:

- 1. **Sentence extraction:** the summary sentence is exactly the same as one of the source sentences.
- 2. **Word extraction:** the summary sentence consists only of words that are present in one of the source sentences.

Table 3.2: Extractiveness of the VIE summaries by year.

Year	**Number of VIE summaries**	**Word extraction**	**Sentence extraction**
2015	8	0.4458	0.1729
2016	24	0.4916	0.2354
2017	147	0.5537	0.2787
2018	573	0.4748	0.2088
2019	325	0.2094	0.07369
2020	204	0.0068	0.0009
2021	126	0.0058	0.0013
2022	49	**0.0**	**0.0**

In order to calculate the sentence extraction and word extraction the VIE summary and the source text had to be broken down into sentences. This was done using spaCy and thereafter the sentence extraction and word extraction was calculated for each sentence in the VIE summary. The results presented in Table 3.2 shows the arithmetic mean for the entire dataset where every data point was given a score calculated by dividing the number of extracted sentences by the total number of sentences in the VIE summary.

As can be seen in Table 3.2 there is a drastic difference in how extractive the VIE summaries are post 2018 compared to the years prior. This problem was at the time noticed and corrected with new leadership and updates to the guidelines for creating VIE Notes. To investigate how this difference in the data affects the training we split the dataset into two. One main dataset which consist of all the data and one subset of this, which we refer to as the abstractive dataset, which only consist of data points where there exists no sentence extraction or word extraction. This filtering was done by calculating the word extraction and sentence extraction for each data point and only including those where the word extraction and sentence extraction were below the threshold of 0.1.

The remaining subset consisted of 644 data points. This subset of the original dataset was further split into training data, validation data and test data with the same proportions as the original dataset. The resulting dataset was therefore 515 (80%) training data, 64 (10%) validation data and 65 (10%) test data.

3.3 Models

The models investigated in this thesis for the task of automatic summarization of VIE summaries were GPT-3 [1] and PRIMERA [27]. Both of these models have been shown to produce state-of-the-art results on summarization tasks and are both based on the transformer architecture. Even though both of them are based on the transformer architecture they differ in many other aspects such as parameter size, architecture (decoder and encoder-decoder) and pre-training objectives.

All of the GPT-3 models were available through the OpenAI API whilst PRIMERA was available both from GitHub and the *transformers* [28] library. The fact that the GPT-3 models were only available through an API posed some problems as it limited the control of the fine-tuning. Another problem was that not all GPT-3 models were available for fine-tuning.

3.3.1 GPT-3

The GPT-3 family of models consisted of many different models and some such as the *gpt-3.5-turbo*, the model which Chat-GPT allegedly used, was not available through the OpenAI API at the start of this thesis work. To properly evaluate whether large language models could be used to generate VIE summaries it was necessary to use the most capable model at the time, which for this thesis work was the *text-davinci-003* model.

The *text-davinci-003* model is an InstructGPT which is a GPT model that has been trained on human feedback to follow instructions [12]. Instructions to the model can be constructed in a lot of different ways and there exist many strategies to construct a prompt for a given task [29], [30]. For this thesis the somewhat arbitrary instruction was chosen:

- **Instruction:** Summarize the following articles:

No major investigation was carried out in regards to the instruction, but rather it was assumed that the instruction would be well aligned with the task at hand. Although the task of producing VIE summaries is not a pure summarization task it is a very similar task and would require very thorough investigation to find the optimal prompt for this setting.

3.3.1.1 Fine-tuning GPT-3

The GPT-3 models that were available for fine-tuning at the time of this thesis were *davinci* (175B parameters), *curie* (6.7B parameters), *babbage* (1.3B parameters) and *ada* (350M parameters). The most capable of these models was the *davinci* model and was thus the model chosen for the fine-tuning as it has been shown to outperform the other models on a wide range of tasks. The reason that the base models were used for fine-tuning is that *text-davinci-003* was not available for fine-tuning. Another important difference between *text-davinci-003* and *davinci* is that *davinci* has only been trained with data up to October 2019. *text-davinci-003* on the other hand has been trained on data up to June 2021.

The *davinci* model also had the constraint that it could only be accessed through the OpenAI API, both for fine-tuning and inference. In the case of fine-tuning the OpenAI API required the input format to be a jsonlines (.jsonl) file as shown below.

```
{prompt: source documents, completion: reference summary}
{prompt: source documents, completion: reference summary}
.
.
.
{prompt: source documents, completion: reference summary}
```

Listing 3.1: Input format for GPT-3 fine-tuning.

The prompt was constructed by concatenating all the source documents with the separator *Article i:* prior to the source document in an attempt to learn that there were different source documents. Furthermore the special string:

```
\n\n##\n\n
```

was added by the recommendation of OpenAI for the model to distinguish between the prompt and the completion.

The fine-tuning had to be carried out through the OpenAI API as the model was not publicly accessible. For this reason there was not complete control of the fine-tuning and the options that could be chosen for the fine-tuning were quite limited. The only hyperparameters that could be controlled were the number of epochs, batch size, learning rate multiplier and prompt loss weight. Furthermore the process of training and doing inference was quite costly, even for a small dataset. The price was $0.03 per 1000 tokens during training and $0.12 per 1000 tokens for inference on the fine-tuned model.

The task of generating VIE summaries was a conditional generation problem. Conditional generation is a problem where output has to be generated given some input, such as summarization, paraphrasing and entity extraction. As this is quite a common task OpenAI had some recommendations to perform a successful fine-tune. First of all it was recommended to add an ending token at the end of completion. This token was " END" and was also used during inference to denote to the model when the generation should stop. Apart from this ending token OpenAI recommended to aim for at least 500 examples, a lower learning rate and 1-2 epochs.

In this thesis we fine-tuned the *davinci* GPT-3 on both the original dataset and the abstractive dataset. Both the models were trained with a learning rate multiplier of 0.05, a prompt loss weight of 0.005 and with a batch size around 0.2% of the training set for 2 epochs. The prompt loss that was mentioned here dictates how much the model should aim to learn the prompt and this was therefore halved from the default prompt loss weight as the prompts in this case were typically long.

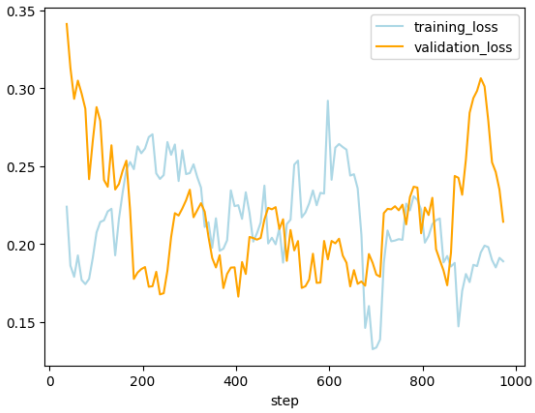

Figure 3.1: Moving average of window 10 for the training and validation loss of GPT-3 trained on the abstractive dataset for 2 epochs.

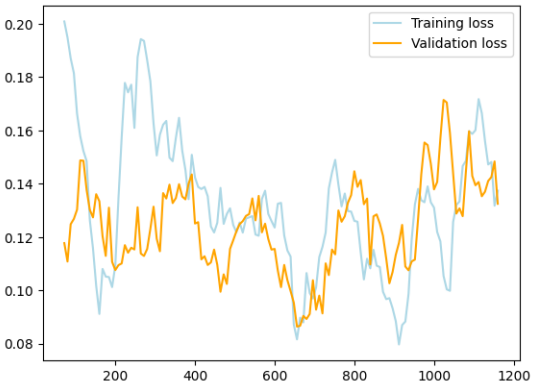

Figure 3.2: Moving average of window 10 for the training and validation loss of GPT-3 trained on the original dataset for 2 epochs.

3.3.2 PRIMERA

As explained in Section 2.1.6 PRIMERA has been trained with a task-specific pre-training objective specifically for the task of multi-document summarization. Thus PRIMERA was chosen as an interesting alternative to GPT-3 as there has been a growing trend to use instruction aligned large language models for all tasks, even though a model such as PRIMERA may perform as well and provides some benefits. Even though PRIMERA is only 447 million parameters compared to 175 billion as in the case of GPT-3 there may not be a need to use very big models for very specific tasks such as summarization.

The benefits that PRIMERA provide are that it is available both on GitHub and the *transformers* [28] library. Furthermore all the details of the architecture and training are known, including the datasets used for training. This yields more control of the model and also gives the opportunity to have the model in-house in cases where the data is sensitive, for legal or privacy reasons.

There exists many fine-tuned versions of PRIMERA on many different datasets, but in this thesis only the pre-trained version of PRIMERA was used in the zero-shot scenario (i.e. it has had no further training apart from the pre-training).

Following the original paper on PRIMERA all the source documents are concatenated into a long sequence before being processed by the model. The global attention was placed on the beginning of sentence (BOS) token as well as the tokens separating these documents (<doc-sep> token).

3.3.2.1 Fine-tuning PRIMERA

PRIMERA was fine-tuned on both the original dataset and the abstractive dataset using the *transformers* [28] library, which provides an effective way to fine-tune open source models.

The actual fine-tuning was carried out using a learning rate of $5 * 10^{-5}$ and a linear decay of the learning rate. Furthermore the weight decay was set to 0.01 and gradient accumulation was used with 16 steps. Gradient accumulation is a technique used to overcome GPU memory limitations and works by running a configured number of steps without updating the model's weights while accumulating the gradients and then using these accumulated gradients to then update the weights.

Furthermore we employed early dropout to prevent overfitting, as can be seen in Figure 3.3 and Figure 3.4 happened already after 100 gradient updates for both the original dataset and the abstractive dataset.

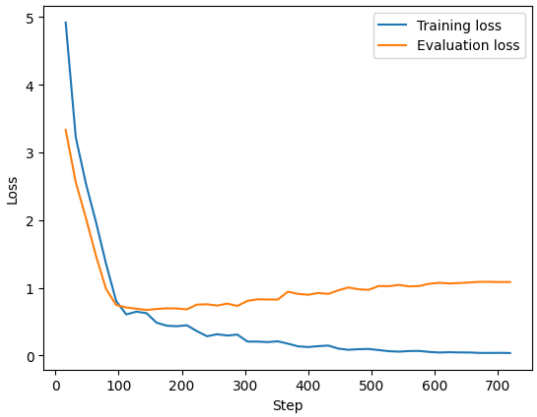

Figure 3.3: Fine-tuning of PRIMERA on the original dataset.

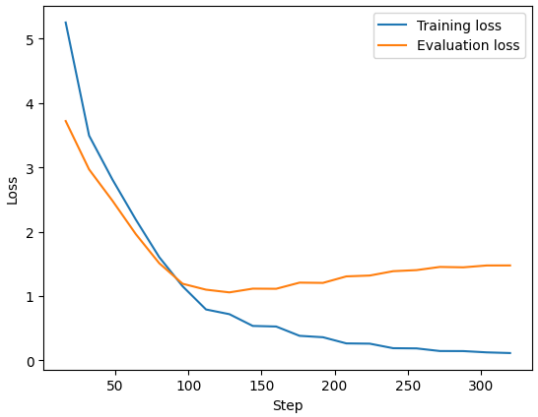

Figure 3.4: Fine-tuning of PRIMERA on the abstractive dataset.

3.4 Decoding Method

Decoding methods in natural language processing involve utilizing the model's probability distribution over the feasible output tokens at each time step to generate a sequence of words. There exist many decoding strategies, with the simplest being to directly sample from the conditional probability distribution.

In this investigation the decoding method that was used in all experiments was to sample from the conditional probability, but with a modified distribution. Instead of sampling directly from the conditional probability distribution one can transform this distribution by altering the *temperature* of the softmax function. If this temperature is increased the likelihood of low probability word increases and if decreased the likelihood of low probability word decreases. Thus this parameter can be used to induce randomness into the model and can be used to make the model more "creative".

Mathematically, the modified softmax with temperature can be expressed as follows:

$$\frac{\exp(\frac{z_i}{T})}{\sum_j \exp(\frac{z_j}{T})}$$

To identify the optimal value for the temperature the ROUGE-1 scores were calculated between the VIE summary and the source document(s) and it was hypothesised that a high ROUGE-1 score between the VIE summary and the source document(s) would imply that the temperature should be set rather low and vice versa.

3.4.1 Temperature Selection

To determine the optimal temperature for the fine-tuned models of GPT-3 we generated summaries with various temperatures for both the abstractive and extractive dataset. It was hypothesized that the ROUGE-1 score between the generated summaries and the source documents should match the ROUGE-1 score between the actual VIE summaries and the source documents.

The ROUGE-1 score between the VIE summaries and the source documents for the abstractive dataset were calculated to be 0.3106. Table 3.3 shows the ROUGE-1 scores between the generated summaries and the source documents at various temperatures for the abstractive dataset. Thus a temperature of 0.9 was chosen in the generation strategy when generating summaries for the model trained on the abstractive dataset. In the case of the original dataset the ROUGE-1 score between the VIE summaries and the source documents were calculated to be 0.3810 and a temperature of 0.6 was therefore deemed the optimal temperature when generating summaries from the model trained on the original dataset.

Table 3.3: Temperature selection abstractive dataset (GPT-3).

Temperature	**ROUGE-1**
0.0	0.3480
0.3	0.3522
0.5	0.3332
0.6	0.3273
0.7	0.3248
0.8	0.3187
0.9	0.3085

Table 3.4: Temperature selection for original dataset (GPT-3).

Temperature	**ROUGE-1**
0.5	0.3989
0.6	0.3834
0.75	0.3654

3.5 Evaluation

This section aims to describe how the evaluation methods used and how the two human evaluations were conducted.

3.5.1 Automatic Evaluation

Automatic evaluation of summarizations can be broadly divided into two different categories: (1) reference-based, i.e. metrics that compare the generated summary against the gold standard summary and (2) reference-free that only rely on the source document [4]. The automatic evaluation conducted in this thesis primarily

relies on reference-based metrics as the dataset consists of pairs with both the gold standard summary as well as the source document(s).

Specifically we calculated the ROUGE-1, ROUGE-2, ROUGE-L, BLEU, BertScore and the number of entities in the gold standard summary that are also present in the generated summary. The first metrics were presented and described previously, but the entity metric require some explanation.

The entity metric worked by extracting all of the entities in the gold standard summary with the help of an in-house provided tool by Recorded Future and then doing the same procedure for the generated summary and looking at how many of the entities from the gold standard summary that made it into the generated summary. This was a particular useful evaluation metric for this case as the entities were of high importance as explained in Section 3.2.1

3.5.2 Human Evaluation

The human evaluation was conducted by four analysts that work in the Insikt group and these were the same throughout the entire human evaluation. We established a set of general criteria for generated text evaluation, as well as a few tailored questions specific to the application of VIE summaries. It was decided to use binary questions (i.e yes or no) for the evaluation metrics. This approach was adopted to mitigate subjective interpretations of the scales, as the perception of values on a scale can vary among different evaluators.. These are the questions used for the evaluation:

- Is the summary fluent and coherent?
- Does the summary answer the 5 W:s (Who, What, Where, Why, When)?
- Does the summary contain redundant content?
- Does the summary accurately portray the underlying facts being presented?
- Are the dates in the summary correct compared to the source?
- Does the summary adhere to the Insikt style guide?
- Is the overall quality sufficient to be used as a VIE Summary?

Furthermore we included two voluntary comment fields give the evaluator a chance to expand on the answers. These were the following criteria the summaries where evaluated on:

- If the overall quality is not sufficient for a VIE Summary, please expand why here
- Please share if you have any additional comment or feedback on the source text, summary or evaluation metrics

The first comment field serves as a follow-up to the question on overall quality. If the evaluator deems the summary of insufficient quality for a VIE Summary, they are asked to expand upon their reasoning in this field. The second comment field

is provided for general feedback. This could include feedback on the source text, summary, or criteria for evaluation.

Recorded Futures Insikt group took on the task to perform the human evaluations of summaries, a highly qualified group of evaluators and a perfect match for the task at hand. For each generated summary, we created a Google Form that included the source article(s), the generated summary, the binary evaluation questions, and the two voluntary comment fields to expand possibly expand the evaluation. To facilitate the process of creating forms and evaluating the results we created Python scripts to create multiple forms rapidly and access answers through the Google API.

3.5.2.1 Evaluation Batches

As we progressed through the thesis work we started dividing the human evaluations into batches. This was done to evenly spread out the workload for the Insikt team but also continuously improve the evaluations and the model. The evaluations was divided into the following batches:

- First batch: 10 summaries with GPT-3 to evaluate initial quality of the summaries and our metrics for human evaluation. This was to function as a proof of concept for the main evaluations for the thesis.
- Second batch: After our first batch we had established the metrics for the human evaluation and adjusted slightly after feedback from the Insikt group. The second batch consisted of 40 summaries using a temperature of 0.4.
- Third Batch: After feedback from the second batch we adjusted the temperature of the GPT-3 model to 0.6. This was done to reduce the extractiveness after feedback from the Insikt team. This batch consisted of 30 summaries.
- Fourth Batch: This batch consisted of 70 summaries. No adjustments were made to the temperature as to have 100 consistent summaries with the same conditions for evaluation.

3.6 Hallucination Evaluation

As was described in an earlier chapter concerning the extractiveness of the data it was of interest to investigate how much of an effect the extractiveness had on the factuality of the generated summaries. It is quite obvious that a very extractive summary can not produce a summary that is not factual since it would be copied from the source, however since the generated summaries are not completely extractive it was of interest to investigate whether the factuality decreased when the training data was more abstractive. Furthermore it was also interesting to investigate whether PRIMERA or GPT-3 were more prone to hallucinations than the other.

Therefore a separate human evaluation was conducted that only focused on determining how prone these models are to hallucinations. For this evaluation the Insikt group was not allowed but instead we, the authors, conducted this evaluation. This

evaluation was conducted by reading through 50 summaries generated in four different settings. One for the fine-tuned PRIMERA on the original dataset, one on the fine-tuned PRIMERA on the abstractive dataset, one for the fine-tuned GPT-3 on the original dataset and lastly one on the fine-tuned GPT-3 on the abstractive dataset. This was done with a temperature of 0.6 for all the models since the evaluation was done to asses how the factuality was affected by the training data and not the temperature.

The evaluation was conducted in a binary way, where each sentence in the generated summary was judged individually whether it was hallucinated or not. The question asked to the evaluators (the authors) was whether this sentence could be inferred given the source text or not. If a sentence was not inferrable given the source text that sentence was deemed as hallucinated.

One exception that was left out of this evaluation was if the dates were hallucinated or wrong. The reason for not including these in this evaluation was that it was detected early that these errors were quite common and the main objective was to determine factual faults and these faults would skew the result in such a way that it would seem as if hallucination was very high, even though the factual faults excluding dates could be quite low.

4

Results

This chapter will present the results from our experiments, through automatic and human evaluation.

4.1 Evaluation

4.1.1 Automatic Metrics

Table 4.1 presents the results of the automatic metrics calculated on the test set on the original dataset for PRIMERA and GPT-3 in the zero-shot setting and for the fine-tuned models. From the automatic metrics it can be seen that there is an improvement in the fine-tuned models compared to the zero-shot setting for both the investigated models.

The automatic metrics for the fine-tuned models are very close, suggesting that automatic metrics are not enough to understand whether one model is better than the other. However, according to Table 4.1 the fine-tuned version of GPT-3 has the best performance and is the model that is further investigated with a human evaluation.

Interestingly it can be seen that the base model of PRIMERA outperforms the zero-shot GPT-3 across the board on the automatic metrics. Most importantly it can be seen that it includes 13% more entities than the zero-shot version of GPT-3.

Table 4.2 presents the results of the automatics metrics calculated on the test for GPT-3 and PRIMERA trained on the abstractive dataset in the zero-shot setting. From this one can see that instead PRIMERA is the best performing model instead of GPT-3, which is a bit surprising. However, the difference is marginal. Furthermore, it is still evident that the fine-tuned versions have better results on the automatic metrics.

Table 4.1: Automatic metrics on original dataset for GPT-3 and PRIMERA with temperature 0.6.

Model	R1	R2	RL	BLEU	BERT	Entity
GPT-3 (zero-shot *text-davinci-003*)	0.47	0.24	0.33	15.44	0.88	0.53
GPT-3 (fine-tuned *davinci*)	**0.59**	**0.40**	**0.46**	**31.29**	**0.91**	0.62
PRIMERA (zero-shot)	0.50	0.30	0.36	21.80	0.90	0.65
PRIMERA (fine-tuned)	0.58	0.38	0.44	29.52	**0.91**	**0.66**

Table 4.2: Automatic metrics on abstractive dataset for GPT-3 and PRIMERA with temperature 0.6.

Model	R1	R2	RL	BLEU	BERT	Entity
GPT-3 (zero-shot *text-davinci-003*)	0.43	0.18	0.28	9.82	0.89	0.45
GPT-3 (fine-tuned *davinci*)	0.51	0.22	0.31	14.48	0.89	0.51
PRIMERA (zero-shot)	0.45	0.26	0.28	12.86	0.87	0.61
PRIMERA (fine-tuned)	**0.52**	**0.29**	**0.38**	**21.35**	**0.90**	**0.64**

4.1.2 Human Evaluation

The results from the human evaluation, presented in Section 3.5.2, are presented in Figure 4.4. The results show that the summaries strike high scores in all categories with the exception of adherence to the Insikt style guide and overall quality. An important note to make is that the overall quality depends on all the other questions in such a way that a summary that is for example not fluent and coherent would also not be of sufficient quality to be used as a VIE summary. In order to properly present the result of the human evaluation each question is presented individually with typical faults the model does.

Is the summary fluent and coherent?

This is one of the metrics that had the highest performance across all questions asked to the evaluators. The most common complaints regarding this question was that the sentences in the summary did not flow together properly or that it had no logical order. An example of criticism on awkward language and flow can be seen in Figure 4.3. The reason why the sentences did not always flow together was that the sentences could be extracted from various parts of the source document(s) and put together without changing these to the extent that this would increase the flow.

Does the summary answer the 5 W:s?

This question requires the summary to include who did it, what happened, where it happened, why it happened and when it happened. However, it was not always the case that all of this could be specified given the source document(s) and the evaluators were in those cases only asked to take into account the ones that could be found in the source document(s). The most common mistake that the fine-tuned version of GPT-3 did was to not include the type of the cybersecurity attack or to not include the severity of the given cyberattack. In most cases that the type

of cybersecurity attack was not included it was only suspected and not confirmed (i.e. it was assumed that it was a given type of attack in the source document(s)). Another common mistake relating to the just mentioned one was that the generated summary did not always include who did it and instead placed more emphasis on who was affected.

GPT Summary

Memorial Hospital at Gulfport notified approximately 1,500 patients of an inadvertent disclosure of information, including patient names and internal (Memorial) encounter numbers, that was discovered during a routine internal audit. The disclosure occurred in connection with the transmission of information within the hospital that was inadvertently sent to a single, external email address as a result of a clerical input error. The error was immediately corrected upon discovery. The information sent by email was encrypted and would require a unique password to open.

Insikt Feedback

The summary does not include the date of the instance, despite it being included in the source. The summary also does not note that the external email address might not even be active, which informs the level of risk associated with this breach. Additionally, the summary uses whole chunks of text directly from the source, which is plagiarism. Finally, the Memorial Hospital at Gulfport won't mean much to most of our clients – we need to provide context as to where this healthcare org is (in the US, I'm assuming), how big it is, if it's linked to a larger healthcare org, etc.

Insikt Summary

Approximately 1,500 Memorial Hospital at Gulfport patients regarding an exposure of their information that was discovered during an internal audit. The disclosure occurred in the transmission of the information within the hospital which the information was sent to an external address as a result of a clerical input error but was immediately corrected upon discovery. The email was encrypted and requires a password to open.

Figure 4.1: Summary of the Memorial Hospital Leak.

Does the summary contain redundant content?

By inspecting Figure 4.4 it can be seen that this was not a major problem for the generated VIE summaries. When inspecting the feedback it could be seen that the

generated VIE summary sometimes repeated itself by repeating the same fact with different phrasing in several sentences.

GPT Summary

Notre Dame de Namur University notified some financial aid applicants that their information may have been compromised when an employee fell prey to a phishing attack. In its notification letter, Henry Roth, the Chief Financial Officer and VP of Administration, writes that the university learned of the possible compromise on May 18. The affected email account contained names, Social Security numbers, and other information provided with financial aid applications. The university offered a complimentary one-year membership of Experianső IdentityWorks.

Insikt Feedback

Though the information is coherent and all of the relevant facts are presented, the writing style is stilted and somewhat informal (mirroring the original text and should be re-phrased into a more formal technical writing style. Also, the date referenced should have the year.

Insikt Summary

An employee of Notre Dame de Namur University fell prey to a phishing attack. The affected email account contained names, Social Security numbers, and other information provided with financial aid applications. The breach occurred on April 23, 2018, but the university learned of it on May 18, 2018. It was not disclosed how many students were affected on the said breach.

Figure 4.2: Summary of Notre Dame University Phishing Attack.

Does the summary accurately portray the underlying facts being presented?

The main criticism for this category highlight summaries where GPT skipped facts that were deemed essential by Insikt. Several summaries miss important facts such as what type of attack or data breach caused the incident. Some summaries presents key information wrong which can lead to more severe consequences. For example a summary mentions data breach where the incident was a mere defacing of a website, no data was extracted or leaked. Figure 4.1 highlights an example where not including facts leads to a higher level of risk for a summary. Here GPT fails to convey the limited severity of the event. In one example GPT mistakedly characterizes the attack to be targeting Linux servers, but the article specifies that the threat actors actually target the the apps that run on top of Linux servers, to eventually get to

the servers underneath.

Are the dates correct?

This category not only concern whether the date presented is actually correct but also if its presented in the correct format. In some instances the dates are completely hallucinated (i.e. no dates are present in the source article) and in some instances the issue lies in the fact that the dates are relative (for example last Thursday instead of 26 May, 2019). Relative dates are typically given in the source articles and therefore GPT-3 would have to transform these dates in order to get absolute dates.

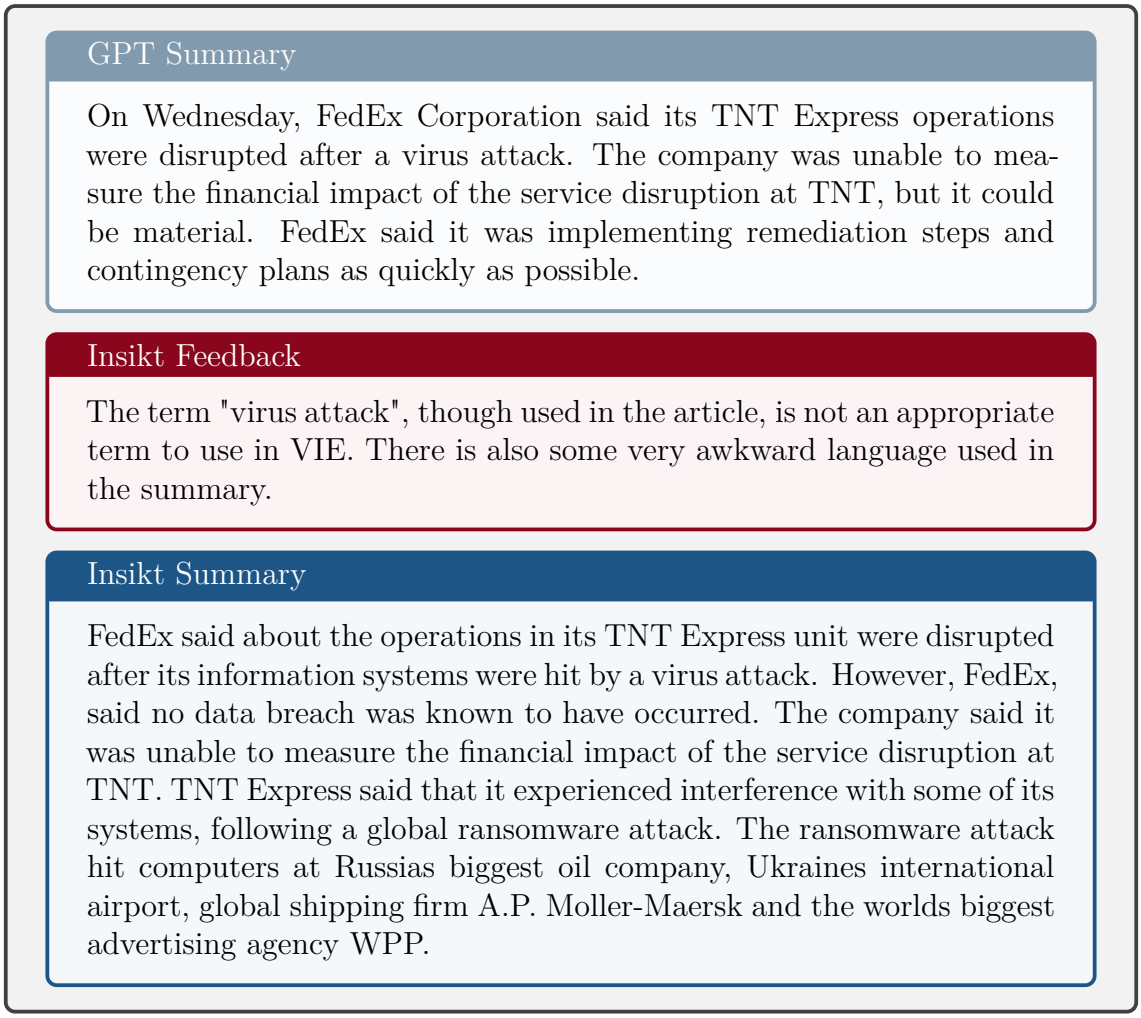

Figure 4.3: Summary of FedEx Virus Attack.

Does the summary adhere to the Insikt style guide?

The Insikt style guide concerns the language used in the summary and in order for a summary to follow the Insikt style guide it should be written in the Insikt voice. Therefore a summary could not adhere to the Insikt style guides in many ways and the generated summaries only managed to follow this style in 41% of the

cases. One problem that occurred was that the source articles were written in a playful tone, which then later transferred over to the summary. This playful tone does not reflect the voice of Insikt and does therefore not adhere to the Insikt style guide. An example where this was the main complaint can be seen in Figure 4.2.

Another, more occurring, problem was that named entities were not properly explained. The source articles may not always explain a named entity in the style that Insikt would, but instead would write, for example: "a five-star hotel in Lukrow". The voice of Insikt would rather explain which hotel it was or at the minimum explain where the city of Lukrow is located geographically. An example of this can be seen in Figure 4.1 where the Memorial Hospital at Gulfport is mentioned without providing additional information about this hospital.

Is the overall quality sufficient to be used as a VIE summary?

This is the lowest performing metric and understandably so since it's is required that all other metrics are satisfied for the overall quality to be sufficient to use as a VIE Summary.

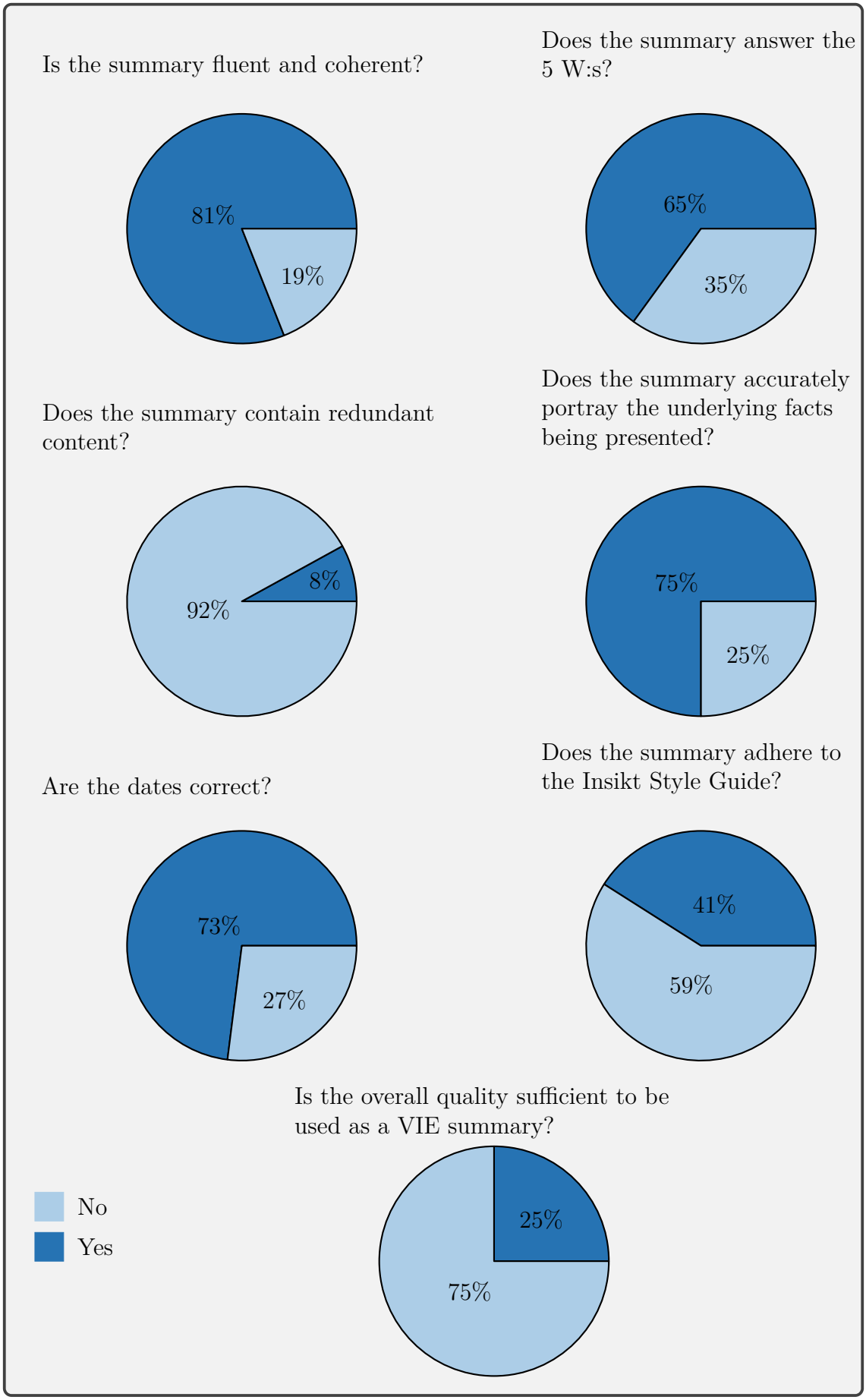

Figure 4.4: Results of human evaluation.

4.1.2.1 Temperature Adjustment

The human evaluation started with 40 evaluations with a temperature of 0.4, but was terminated early due to complaints regarding plagiarism. To counteract the plagiarism the temperature was changed to 0.6 to decrease plagiarism, which can be seen in Table 4.5.

In Table 4.3 the results of the initial 40 evaluations can be seen as well as the actual human evaluation of 100 evaluations. It can be seen that this increase lead to a more fluent and coherent summary at the cost of more redundant content as well as a slight decrease in factual correctness and correct dates. However, the overall quality did not change.

Table 4.3: Human evaluation on GPT-3 for temperature 0.4 vs human evaluation for temperature 0.6.

Metric	**Temp 0.4**	**Temp 0.6**
Is the summary fluent and coherent?	72%	81%
Does the summary answer the 5 W:s?	65%	65%
Does the summary contain redundant content?	2%	8%
Does the summary accurately portray the facts being presented?	77%	75%
Are the dates correct?	87%	73%
Does the summary adhere to the Insikt style guide?	40%	41%
Is the overall quality sufficient to be used as a VIE summary?	25%	25%

4.2 Extractiveness/Factuality

The previous sections discussed the human evaluation of GPT-3 and the comparison of its performance with PRIMERA using automatic metrics. In this section, additional findings will be presented to explore the relationship between extractiveness and factors such as temperature and training data. Furthermore, the impact of using less extractive training data on the factuality of the generated summaries will be examined.

4.2.1 Extractiveness

Table 4.4 and Table 4.5 presents the extractiveness of the fine-tuned versions of GPT-3 and PRIMERA when trained on the different datasets. Inspecting the extractiveness of the models trained on the original dataset it can be seen that the extractiveness is very high and also that GPT-3 is more extractive, which suggests that it may more easily fit to the pattern of extractiveness when the training data is extractive.

Table 4.4: Word extraction and sentence extraction for the abstractive dataset at various temperatures.

Model	**Word extraction**	**Sentence extraction**
GPT-3 (abstractive dataset, temperature 0.3)	0.506	0.277
GPT-3 (abstractive dataset, temperature 0.6)	0.163	0.075
GPT-3 (abstractive dataset, temperature 0.9)	0.047	0.001
PRIMERA (abstractive dataset, temperature 0.3)	0.038	0.018
PRIMERA (abstractive dataset, temperature 0.6)	0.081	0.033
PRIMERA (abstractive dataset, temperature 0.9)	0.113	0.053

Table 4.5: Word extraction and sentence extraction for the original dataset at various temperatures.

Model	**Word extraction**	**Sentence extraction**
GPT-3 (original dataset, temperature 0.3)	0.6840	0.3924
GPT-3 (original dataset, temperature 0.4)	0.6137	0.3526
GPT-3 (original dataset, temperature 0.6)	0.5962	0.3172
GPT-3 (original dataset, temperature 0.9)	0.3064	0.1464
PRIMERA (original dataset, temperature 0.6)	0.5306	0.2491

Inspecting Table 4.4 it can be seen that the extractiveness is significantly lower with more abstractive training data, which indicates that the the extractiveness of the training data has an enormous impact on the extractiveness of the generated summaries.

One can also see the effect that the temperature seems to have on the extractiveness. Even though the temperature seems to have some effect on the extractiveness of the generated summaries it does not seem to have the strong effect as the training data.

4.2.2 Factuality

The results of the human evaluation presented in Section 3.6 is presented in Table 4.6. The average amount of hallucinated sentences for the GPT-3 fine-tuned on the original dataset corresponds very well to the results of the human evaluation and it can also be seen that the performance of PRIMERA and GPT-3 seem to be quite close. Training on the abstractive dataset clearly resulted in a higher occurrence of hallucinated sentences. Nonetheless, it is likely that further experiments adjusting the models temperature could mitigate or reduce hallucinations in the summaries.

However, the results here highlight the important factuality-extractiveness trade-off that needs to be considered when generating summaries.

Table 4.6: Hallucinated sentences.

Model	Average hallucinated sentences
GPT-3 (fine-tuned *original dataset*)	0.127
GPT-3 (fine-tuned *abstractive dataset*)	0.232
PRIMERA (fine-tuned *original dataset*)	0.041
PRIMERA (fine-tuned *abstractive dataset*)	0.194

5

Discussion

This chapter discusses the results presented in Chapter 4, starting with a discussion of the results of the human evaluation and then a discussion of the trade-off between extractiveness and factuality is discussed.

5.1 Human Evaluation

The main objective of this thesis was to evaluate how well GPT-3 and PRIMERA perform in the task of generating higher quality summaries such as VIE summaries and whether fine-tuning of these models will have an improvement on the generated summaries. In order to conduct such an experiment one has to perform some kind of human evaluation, as automatic metrics have been shown multiple times not to correlate with human judgement.

However as this thesis did not have the resources to conduct several human evaluations it was decided to only conduct one such evaluation on the fine-tuned davinci version of GPT-3. The reason was that this was deemed as the most capable model of these and was expected to perform the best on this task. This has the drawback that the comparison is not complete, as a complete comparison would require at the minimum two human evaluations (one for fine-tuned PRIMERA and one for fine-tuned GPT-3). Thus the second human evaluation only concerning factuality aimed to provide a middle-ground, without being a complete evaluation.

Given additional resources, we would have preferred to conduct further investigations and human evaluation for comparisons involving various aspects. These include evaluating the performance of GPT-3 compared to PRIMERA, exploring different settings of GPT such as zero-shot, few-shot, and fine-tuned, experimenting with different temperature settings, and utilizing datasets with varying levels of extractive training data. Expanding our research in these areas would provide deeper insights into the capabilities and performance of different models and variations, and presumably led to the best possible VIE summary.

5.1.1 Results Of The Human Evaluation

The main challenge of this thesis was not merely summarizing information, but rather replicating a specific voice. The challenge extended beyond the summarization task itself and required producing a higher quality summary, capturing the

unique voice and style characteristic of the Insikt Group's Validated Intelligence Events (VIEs). Insikt Group describes their own VIEs as delivering high-quality information. As such, the VIEs are not run-of-the-mill news summaries or simple compressions of information. Instead, they represent specific pieces of intelligence with a distinct voice and a level of quality that sets them apart. In order to properly evaluate the generated summaries based on this criteria, we included additional metrics to consider the specific style and voice of Insikt. Ultimately, these metrics received low scores in the human evaluations, further emphasizing the challenge of replicating a specific voice and generating higher quality summaries beyond mere informational content.

Interestingly enough, we can observe how the performance on specific metrics changes as the temperature was altered. This comparison can be seen in Table 4.3. However, the evaluation is incomplete due to the limited number of evaluations, with only 40 evaluations at temperature 0.4 and 100 evaluations at 0.6. Even though it is incomplete it's interesting that such a small change in temperature could lead to a 9% increase in fluency and coherence, a 6% increase in redundant content and 14% decrease in correct dates. The overall quality stayed the same but it can be seen that the temperature seems to alter various metrics differently. The fact that the generated VIE summary is more fluent and coherent is most likely due to a decrease in extractiveness which can also be seen in Table 4.5. Intriguingly this improvement in fluency and coherence is accompanied by an increase in redundant content and a decrease in accurate dates, potentially indicating the presence of hallucination.

The main metric (Is the overall quality sufficient to be used as a VIE summary?) of the evaluation of the generated VIE summaries for the fine-tuned davinci GPT-3 model at temperature 0.6 showed quite low results with only 25%. GPT-3 has previously been shown to perform very well on general summarization tasks, but not a lot of research has been done in cases where the generated summary needs to be domain specific, query-oriented (see Section 2.1.1) and also follow strict guidelines regarding the style. Therefore this result may not actually be low when these extra criteria are taken into consideration.

When analyzing the results of the human evaluation it can be seen that the most common complaint present in the human evaluation was of plagiarism. In this context, this means that there were extracted sentences or words from the source document(s). This extractiveness has been shown to be present due to the extractiveness of the training data, which can be seen in Table 4.5. Therefore this complaint is not surprising, but may not have been a complaint if the generated VIE summaries were to be used prior to 2018 (see Table 3.2).

Further problems that were present were that the generated VIE summaries did not adhere to the Insikt style guide. One possibility why this especially was a problem, apart from extractiveness, is that the style of the summary seems to be partly dependent on the source document(s). If the source document(s) follow a style that is similar to the Insikt style guide one can expect that the generated summary will also follow this style, but as the source document(s) start to diverge from this style of writing then it seems as the generated summary will too. This is something that

could be further investigated to determine how dependant the model is on the source document(s), as in the best case scenario the model would not be dependant at all on the style of the writing in the source document(s).

5.2 Extractiveness And Factuality

This thesis work also explored how the extractiveness of the generated VIE summaries were affected by changing the temperature and the training data. The extractiveness of the models seems to be very dependant on the training data and one can clearly see that these large language models easily pick up this pattern of extracting sentences or part of sentences from the source document(s).

One interesting aspect of this is that the fine-tuned davinci GPT-3 model trained on the original dataset seems to pick up this pattern easier than the fine-tuned PRIMERA trained on the original dataset. This is somewhat surprising given that the pre-training of PRIMERA consists of extracting salient sentences from the source document(s) and it was thus expected that this model would be less extractive than GPT-3.

.

6

Conclusion

From the discussion carried out it is clear that a fine-tuned version of GPT-3 was not able to completely automate the task of producing VIE summaries. However, it was shown to tremendously help the analysts who are currently writing these in cutting down the production time. This has not been verified completely, but according to one of the analysts it was expected to cut down the production time by a factor of four. Furthermore, just by going by extractiveness, factuality and automatic metrics it was shown that PRIMERA may perform very similar to GPT-3 and shows that it is a viable option to use a smaller model in such cases.

Whether a more successful fine-tuning would have been possible if the summaries in the dataset was more similar to one another in term of extractiveness and writing style is still of interest. Unfortunately this was not the case throughout this thesis and even though we did a separate fine-tuning on very filtered data it may have been a too small dataset. The separate discussion about the extractiveness of the fine-tuned models with regards to the extractiveness of the training data clearly shows that these models will be very extractive if the training data is very extractive.

Bibliography

[1] T. Brown, B. Mann, N. Ryder, *et al.*, "Language models are few-shot learners," *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.

[2] A. Chowdhery, S. Narang, J. Devlin, *et al.*, *Palm: Scaling language modeling with pathways*, 2022. arXiv: 2204.02311 [cs.CL].

[3] M. Allahyari, S. Pouriyeh, M. Assefi, *et al.*, "Text summarization techniques: A brief survey," *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 10, 2017. DOI: 10.14569/IJACSA.2017.081052. [Online]. Available: http://dx.doi.org/10.14569/IJACSA.2017.081052.

[4] T. Goyal, J. J. Li, and G. Durrett, "News summarization and evaluation in the era of gpt-3," *ArXiv*, vol. abs/2209.12356, 2022.

[5] Y. Liu and M. Lapata, "Text summarization with pretrained encoders," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Hong Kong, China: Association for Computational Linguistics, Nov. 2019, pp. 3730–3740. DOI: 10.18653/v1/D19-1387. [Online]. Available: https://aclanthology.org/D19-1387.

[6] P. Bhattacharya, K. Hiware, S. Rajgaria, N. Pochhi, K. Ghosh, and S. Ghosh, "A comparative study of summarization algorithms applied to legal case judgments," in *Advances in Information Retrieval: 41st European Conference on IR Research, ECIR 2019, Cologne, Germany, April 14–18, 2019, Proceedings, Part I 41*, Springer, 2019, pp. 413–428.

[7] S. Dutta, V. Chandra, K. Mehra, S. Ghatak, A. K. Das, and S. Ghosh, "Summarizing microblogs during emergency events: A comparison of extractive summarization algorithms," in *Emerging Technologies in Data Mining and Information Security: Proceedings of IEMIS 2018, Volume 2*, Springer, 2019, pp. 859–872.

[8] C. D. Manning, H. Schütze, D. M. Powers, and C. C. Turk, "0.2. history of natural language processing 7," *Natural Language Processing*, p. 7, 1990.

[9] G. Sharma and D. Sharma, "Automatic text summarization methods: A comprehensive review," *SN Computer Science*, vol. 4, no. 1, p. 33, 2022.

[10] E. Hovy and C.-Y. Lin, "Automated text summarization and the summarist system," UNIVERSITY OF SOUTHERN CALIFORNIA MARINA DEL REY INFORMATION SCIENCES INST, Tech. Rep., 1998.

[11] B. Zoph, C. Raffel, D. Schuurmans, *et al.*, "Emergent abilities of large language models," *TMLR*, 2022.

[12] L. Ouyang, J. Wu, X. Jiang, *et al.*, "Training language models to follow instructions with human feedback," in *Advances in Neural Information Processing Systems*, S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, Eds., vol. 35, Curran Associates, Inc., 2022, pp. 27 730–27 744. [Online]. Available: `https://proceedings.neurips.cc/paper_files/paper/2022/file/b1efde53be364a73914f58805a001731-Paper-Conference.pdf`.

[13] A. Vaswani, N. Shazeer, N. Parmar, *et al.*, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[14] K. Cho, B. van Merrienboer, Ç. Gülçehre, *et al.*, "Learning phrase representations using rnn encoderdecoder for statistical machine translation," in *Conference on Empirical Methods in Natural Language Processing*, 2014.

[15] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[16] J. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," *ArXiv*, vol. abs/1607.06450, 2016.

[17] R. Xiong, Y. Yang, D. He, *et al.*, "On layer normalization in the transformer architecture," in *International Conference on Machine Learning*, PMLR, 2020, pp. 10 524–10 533.

[18] I. Beltagy, M. E. Peters, and A. Cohan, "Longformer: The long-document transformer," *arXiv:2004.05150*, 2020.

[19] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, *et al.*, "Language models are unsupervised multitask learners," *OpenAI blog*, vol. 1, no. 8, p. 9, 2019.

[20] R. Child, S. Gray, A. Radford, and I. Sutskever, "Generating long sequences with sparse transformers," *ArXiv*, vol. abs/1904.10509, 2019.

[21] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," in *Text summarization branches out*, 2004, pp. 74–81.

[22] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: A method for automatic evaluation of machine translation," in *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, 2002, pp. 311–318.

[23] T. Zhang*, V. Kishore*, F. Wu*, K. Q. Weinberger, and Y. Artzi, "Bertscore: Evaluating text generation with bert," in *International Conference on Learning Representations*, 2020. [Online]. Available: `https://openreview.net/forum?id=SkeHuCVFDr`.

[24] J. D. M.-W. C. Kenton and L. K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of naacL-HLT*, vol. 1, 2019, p. 2.

[25] D. Khashabi, G. Stanovsky, J. Bragg, *et al.*, "Genie: Toward reproducible and standardized human evaluation for text generation," in *Conference on Empirical Methods in Natural Language Processing*, 2021.

[26] C. van der Lee, A. Gatt, E. van Miltenburg, and E. Krahmer, "Human evaluation of automatically generated text: Current trends and best practice guidelines," *Computer Speech & Language*, vol. 67, p. 101 151, 2021.

[27] W. Xiao, I. Beltagy, G. Carenini, and A. Cohan, "Primera: Pyramid-based masked sentence pre-training for multi-document summarization," in *Annual Meeting of the Association for Computational Linguistics*, 2021.

[28] T. Wolf, L. Debut, V. Sanh, *et al.*, "Transformers: State-of-the-art natural language processing," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, Online: Association for Computational Linguistics, Oct. 2020, pp. 38–45. DOI: 10.18653/v1/2020.emnlp-demos.6. [Online]. Available: https://aclanthology.org/2020.emnlp-demos.6.

[29] S. Arora, A. Narayan, M. F. Chen, *et al.*, "Ask me anything: A simple strategy for prompting language models," *arXiv:2210.02441*, 2022.

[30] L. Reynolds and K. McDonell, "Prompt programming for large language models: Beyond the few-shot paradigm," in *Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems*, 2021, pp. 1–7.

A

Appendix 1