



**Universiteit
Leiden**
The Netherlands

Opleiding Informatica & Economie

Benchmarking Large Language Models
for domain-specific summarization

Rens Piké, s2780097

Supervisors:

Peter van der Putten & Guus Ramackers

BACHELOR THESIS

Leiden Institute of Advanced Computer Science (LIACS)

www.liacs.leidenuniv.nl

10/07/2023

Abstract

Building generative AI capabilities may require a mix of different foundation models based on expected performance, cost, ability to leverage proprietary data and other pre-determined requirements. This study focusses on summarization as a specific use case, within the context of a software company developing generative AI capabilities. A variety of publicly available services, such as GPT-3.5-turbo and GPT-4, are benchmarked against open source models such as Pythia and MPT-7B, which are fine-tuned on company specific data. This research reports the results on open data sets as well as company proprietary data. Whilst the open models are not yet up to par in terms of quality, these results are useful for specific use cases, where use of public services may not be cost effective or appropriate given the nature of the data.

Contents

1	Introduction	1
1.1	The situation	1
1.2	Thesis overview	1
2	Background	2
2.1	Generative AI	2
2.1.1	Training	2
2.1.2	Input	2
2.1.3	Output	2
2.2	Pegasystems Inc.	2
2.3	Use case	3
2.3.1	Summarization	3
2.3.2	Large Language Model	3
2.3.3	Tokens	4
2.3.4	Availability	4
2.4	Other research	4
2.4.1	OpenAI	4
2.4.2	Generative AI performance	5
2.4.3	Faithful summary creation	5
2.4.4	Prompt selection	5
3	Methods	5
3.1	Models	5
3.1.1	FLAN-T5-XXL	5
3.1.2	GPT-4	6
3.1.3	GPT-3.5-turbo	6
3.1.4	Pythia	6
3.1.5	MPT-7B	7
3.2	Model evaluation	7
3.2.1	ROUGE-test	7
3.2.2	Best summary	8
4	Experimental set up	8
4.1	Preliminary experiment	9
4.2	News data set	9
4.3	Pega documentation data set	10
4.4	News- and Pega documentation data set	10
5	Results	11
5.1	Preliminary experiment	11
5.1.1	Results	11
5.1.2	Best summary	12
5.2	News data set	12
5.2.1	Results	12

5.2.2	Best summary	14
5.3	Pega documentation	15
5.3.1	Results	15
5.3.2	Best summary	16
6	Discussion	17
6.1	Reflection on results	17
6.2	Limitations	17
6.3	Future work	18
7	Conclusions	19
	References	21
	Appendix	22

1 Introduction

Artificial intelligence (AI) is a subject that is becoming more and more common knowledge and with that, the rise of AI technology. [5] Generative AI is one of those subjects that is talked about more and more. It can predict anything by analyzing data and the input, which will help to generate the output. The possibilities for implementation that arise when talking about generative AI are endless, which causes companies to look into it to see opportunities for change. It could be implemented internally or into the services for the customer to use.

This research is supported by an internship at Pegasystems Inc. to give access and insight in their current procedures. [24] This thesis is inspired by a previous assignment done for Pegasystems Inc. for a course in the Computer Science & Economics curriculum. In this assignment, possibilities for the implementation of generative AI had to be found and explained. This created the gateway into this research to further examine these possibilities and to see how extra research could contribute to the current way of working. This can give a more concrete view of how the possibilities could be implemented and of influence to the current processes at Pegasystems Inc.

1.1 The situation

With this rise in developments of AI, commercial businesses want to know if it is possible to implement it into their business. Pegasystems Inc, in short Pega, is such a business, which already is using AI in their services. [24] To further enhance their products it is necessary to know which of these new models are worth to look into given a specific use case. This needs to be tested and multiple AI models have to be taken into consideration. Multiple requirements have to be taken into consideration, as accuracy, quality and costs play a role. Furthermore, the purpose of the model has to be taken into account depending on the use case, for example private models are considered better for personal data, as these models do not share the data with centralised public services. To be able to decide if the current practices of Pega are able to compete with the competitors, to use the following research question is formulated:

How could the summarization capabilities of various Large Language Models be compared on summarization tasks for both public- and domain specific data and models?

The goal of this thesis is to be able to show how Large Language Models can be tested and if the process at Pegasystems Inc. needs to be changed. The aim of the testing is to be generic, but the testing might involve the use of non-public data, which will not be accessible for further usage. This thesis will describe the effectiveness of the testing to be able to use it as a set up for a final recommendation.

1.2 Thesis overview

The remainder of this thesis is structured as follows; Section 2 includes the definitions and background of often used terms and subjects, as well as related work; Section 3 explains the selection of Large Language Models and the evaluation metrics; Section 4 describes the experimental set up for the experiments that were carried out; Section 5 shows and explains the results of the experiments; Section 6 includes a discussion on the subject and the possibilities for future research; Section 7 concludes this thesis.

2 Background

The following section gives extra information on what Generative AI and Large Language Models are, what Pega does as a company and what kind of use case will be carried out in this research.

2.1 Generative AI

Generative AI is a term that is used a lot for models that can generate something, this can be text, images or speech for example. It might sound simple at first, but the process of making such a model work properly to give the right answers, takes a lot of time and money.

2.1.1 Training

When looking at Large Language Models, they firstly have to go through training. It will be given a lot of data to train itself a language and its structure. This can be done by using a neural network architecture, which gives the possibility to train with data from multiple resources. The pre-training step can be a very expensive process, depending on the goal of the model and the amount of data it will be trained on, this is typically only done by the foundation model developer. The model can then be trained further through fine-tuning and reinforcement learning from feedback. This is typically also done by the foundation model provider, but end users as Pega can further fine-tune the model. This will be discussed in more detail in Section [2.3.2](#).

2.1.2 Input

After the training of the data, the model will need to know what kind of output it has to deliver and how to format it. This is done in through prompt engineering, where the developers can tell the system how to perform the request given by the users. This can be a question that needs an answer or an assignment to write or create something, but it can also be additional context or formatting requirements. The model will break the prompt into smaller pieces, tokens, to be able to analyze what it requested. By doing this, the model can acknowledge the prompt and predict what is wanted. This is based on the training done by the model, which gives it the possibility to recognize patterns in the text.

2.1.3 Output

To be able to answer the question or assignment, the model uses statistical patterns to generate the most likely next word in the sentence. This is a process that is learnt during the training, by analysing the text data and finding connection of words in sentences. This way, the model will understand the user input and will know what to do next (answer the question or create the assignment). The output that is generated will be formulated in a way that it will look like the user is communicating with another human being (this is not the case in image creation of course).

2.2 Pegasystems Inc.

Pegasystems Inc. (Pega) was founded in 1983 in the United States by Alan Trefler and is a company that develops software. [\[24\]](#) Its catchphrase is 'A low-code platform for AI-powered decisioning and

workflow automation' and with that they expanded to 40 offices worldwide with 6500 employees. [23] It offers software for multiple goals, like Intelligent automation, Customer service and Sales automation. [25]

In two recent press releases by Pega, they announced they are going to implement generative AI into their services. This will give the users the opportunity to use AI in their daily processes to speed them up. Tasks like 'create a report' or 'build this application' will help users to automate their daily work and work more efficiently. [27][26] Furthermore, implementing generative AI into Pega's services will also give the possibility to summarize documents or create customer engagement content. This could contribute to providing better service or recommendations to the customer. [1]

2.3 Use case

There are multiple different use cases, which are not all in the chatbot type of style, as given in the section above. The determination of the best model depends on the use case and on different factors. Examples of these factors are quality of the output, cost or the possibility to use confidential data without sharing the content. In short, multiple models could be used, but how to determine the best model for the use case?

2.3.1 Summarization

A key task for Generative AI is summarization, which can be used for multiple purposes. This is a subject that is very relevant for Pega, as the implementation of summarization can lead to better services. Think about customer service, where previous conversations can be summarized to give employees a quick view on the subject and a good understanding of the problem. AI can look through conversations, transcripts and other documents to make sure the employees are able to give better service to customers in a more efficient way. [1] Furthermore, summarization could also be used for technical reports, where information is given on updates or some instructions for performing certain tasks.

This thesis will therefore focus on the summarization capabilities of models accessible to Pega, see section 3.1 below for a definite selection of the models that will be tested. These models will be tested by running them through the ROUGE test, which will indicate if the summary that is created by the model is good compared to (human) reference. [2] It is the main test for the evaluation of text summarization. For some models, evaluation results are available on public data sets. For Pega it is also relevant to know whether these results compare to results obtained on domain-specific data sets and use cases. This way it will be possible for Pega to make informed choices on what models to use for specific use cases.

2.3.2 Large Language Model

A Large Language Model (LLM) is a large version of a language model, a model that is the base to communication, existing of words, grammar rules and meanings. These are used to let the AI create new things. The large part of LLM comes from the fact that these language models are trained on a lot of data, which helps the AI model at doing its tasks successfully. The amount of data that is used by a language model to become a Large Language Model, is not fixed, but it is often at least one hundred million tokens. [12]

2.3.3 Tokens

Tokens are the variables that are used to train language models. To make a model better at its tasks, it is necessary to be trained on more data, which means that there are more tokens needed. As stated above, millions to hundreds of billions of tokens are used for a Large Language Model, however this is a small amount compared to most models in the industry. [12] OpenAI's GPT-3 model has 175B parameters the original version has been trained on 499 billion tokens and Google's PaLM is trained on 540 billion tokens. [7] [17]

2.3.4 Availability

All Generative AI models that are released by companies or individuals must have a known license to set rules about copying, usage, sharing and reproduction of the code. This license states what the user of the model code is allowed to do. The license can be open-source, which can also be for non commercial use only. When the license is open-source non commercial use only, it is less valuable model for Pega, as it can be used for benchmarking, but not for production use. [34]

Understanding the correct license constraints can be a tricky objective, because a lot of models that are released are created on or with another one. Here an example to clarify this process. Alpaca is a LLM made by Stanford University students has the Apache 2.0 license. [32] This license states that users are allowed to use, copy and modify code with this license, but are asked to give credit to the original developers. [10] So it is expected that anyone can use this model, under one condition, giving credit to the authors. However, this is not what is true. This is because Alpaca is made on LLaMa by Meta, which is created for non commercial use only. [32] [33] This causes Alpaca to be for non commercial use only too.

2.4 Other research

To show the relevancy of this research, other research related to this subject is given below.

2.4.1 OpenAI

OpenAI [19] is one of the leading companies developing generative AI models. They recently released their top of the line model, GPT-4, a 'large multimodal model' that out-performs almost all existing models available. [22] This model is able to use different types of input, in this case text and image, and generate text as output.

In the research paper related to the release of GPT-4, GPT-4 is compared to other models in multiple benchmarks, like simulating exams or 'benchmarks designed for machine learning models'. [21] [22] In the latter, six benchmarks are used to determine the performance of the GPT-4 model. In five of the six benchmarks, GPT-4 performs much better than the current top-scorer, which shows the capabilities of this model. This paper will test and use GPT-4 as a reference to be able to see its abilities.

However, GPT-4 is also not perfect, like the other models. As stated on the web page of the release by OpenAI, GPT-4 'has similar limitations as earlier GPT models'. [22] This means it still creates answers that do not make sense and makes mistakes when trying to give argumentation on a subject.

2.4.2 Generative AI performance

To give users more time for important tasks or to just make them work more efficiently, it is useful to implement generative AI to summarize context or documents. This summarization needs to be done in a way that people using it or reading it can still understand it. Meaning it needs to be natural language and containing the key points from the original text. Furthermore, there is stated that GPT was able to give similar answers when set side by side to human replies. [13]

2.4.3 Faithful summary creation

The introduction of ChatGPT caused a rise in interest into chatbot type of generative AI, as it is capable of performing multiple tasks very well. However, it is not performing well in each task, as given by Zhang, H., Liu, X. and Zhang, J., the ability of ChatGPT in extractive summarization is 'inferior to existing supervised systems in terms of ROUGE scores.' [35] By using different public data set benchmarks, the ability of ChatGPT in summarization is shown.

2.4.4 Prompt selection

To be able to compare the experiments later on in this research, it is necessary to assert equal prompts for the different models. The prompt usage for different texts should be the same to be able to determine any outliers in the results of a specific model. Furthermore, when comparing the results of multiple models, it is crucial to use the same prompt. This is because the prompt decides what kind of summary is created by the model. If these 'settings' are not equal, different results will show and no recommendations can be done to Pega.

Prompts consist of multiple aspects and changing them around can 'change the accuracy from near chance (54%) to near state-of-the-art (93%)' [36]

3 Methods

In this section, more information on the different models that will be used for the different experiments will be given. Moreover, the evaluation methods for the different experiments will be explained thoroughly.

3.1 Models

In each experiment, different Large Language Models are used and to be able to understand them, additional information is provided below. An overview of the details is given at the end in Table 1.

3.1.1 FLAN-T5-XXL

The first model that is used is FLAN-T5-XXL. This is a fine-tuned version of the original version T5, which is created by Google as a text-to-text transfer transformer. [30] It was released in early 2020 in multiple versions depending on the amount of parameters. These were 60M, 220M, 770M, 3B and 11B which were indicated by Small, Base, Large, XL and XXL. [29]

All these versions were later fine-tuned to give them the ability to do extra tasks as an upgrade, which gave them the prefix FLAN. [6] The FLAN models are all open sourced by Google and for

commercial use, so should Pega want to implement a FLAN model in its services, it is allowed to do so. To limit the scope of this research, FLAN-T5-XXL (the 11B parameter model) is the only model that is tested from the available FLAN-T5 models.

3.1.2 GPT-4

As mentioned in section 2.4, GPT-4 is the most up-to-date model by OpenAI. [22] It was released on the 13th of March 2023 and it is capable of performing on a higher level than any other model launched by OpenAI. This is because GPT-4 is build with additional models that are able to help its reasoning abilities and it is given a wider knowledge base. [21][18]

For the simple tasks, the difference between GPT-4 and GPT-3.5 is not significant, but the power of GPT-4 becomes visible when more complex problems are executed. This can be seen in the different results for the simulated bar exam for example. GPT-4 scores 298/400, which places it in the 90th percentile, compared to GPT-3.5 with 213/400 or the 10th percentile. [21][22]

3.1.3 GPT-3.5-turbo

GPT-3.5-turbo is another model from OpenAI which is a predecessor of GPT-4. [19] It was released on March 1st of 2023 and is an improved version of GPT-3.5. It can create code and text and is a model with 175B parameters. However, it is one of the least expensive models of the GPT family that is not fine-tuned. [18] The pricing of models is per 1K tokens, where GPT-3.5-turbo only costs \$0.0015 for processing input and \$0.002 for creating output. For comparison, GPT-4 costs \$0.03 for input and \$0.06 for creating output per 1K tokens, which means it is more than 20x as expensive than GPT-3.5-turbo. [20]

It was initially optimized for chat usage but other types of tasks can be done effectively.

3.1.4 Pythia

Pythia is a open sourced Large Language Model released by EleutherAI on February 13th of 2023. [9] It is trained on 'The Pile', which is a dataset with 800 GB of English texts for the training of Large Language Models. [11]

Pythia is not one model, it is a collection of 16 different models, with 8 different sizes, from 70M parameters to 12B parameters. These sizes each occur twice, because EleutherAI trained each size on the Pile and the Pile 'deduped'. [4] The 'deduped' Pile is the dataset the Pile, fine tuned to remove the duplicates in it, which helps with the training of the model. [8]

This is one of the many internally models that Pega has running. This model is used by the Machine Learning team to perform multiple tasks. For this reason the model is fine-tuned on documentation of Pega, where it got 8k different question and answers related to Pega. Inside Pega there is a GPT powered chatbot, which is able to answer questions about Pega documentation. The usage of this chatbot results into a corpus of questions and answers about Pega. This is given to the internal models to see how they perform on this data and the ability to answers questions on Pega subjects. It is still able to do other tasks than answering questions and therefore this model is used in this research.

Model	Parameters	Number of tokens
FLAN-T5-XXL	11B	34B
Pythia	12B	300B
MPT-7B	7B	1T
GPT-3.5-turbo	175B	499B
GPT-4	not published	not published

Table 1: Overview of the models

3.1.5 MPT-7B

MPT-7B is another open sourced Large Language Model, released by MosaicML available for commercial usage. [16] It is trained on 1T tokens, consisting of text and code and the performance is similar to LLaMa-7B. To compare, Pythia is trained on the Pile or 300B tokens, so MPT-7B is trained on more than 3x as many tokens.

This model is another model that Pega are able to run internally for their own usage. This model is also trained on documentation of Pega, on the 8k Pega question and answering and the 63k MPT instruct data set from MosaicML. With the latter the model is able to perform multiple short-form tasks. [16]

Both Pythia and MPT-7B were fine-tuned before the start of this thesis, as these models were already running internally. Access to these models was given to see their performance compared to the other models.

3.2 Model evaluation

The summaries that are created by the models are evaluated in multiple ways, the methods for these evaluations are described in the following subsections.

3.2.1 ROUGE-test

The ROUGE test is divided into three sections, ROUGE-1, ROUGE-2 and ROUGE-L. In ROUGE-1, the summaries are compared on a uni-gram level, which means that each word in both summaries will be compared. In the ROUGE-2 section, the summaries are compare on a bi-gram level, which means that both summaries will be split up into a section of two consecutive words. In the final section, ROUGE-L, the summaries are compared on LCS, or Longest Common Subsequence. [14] Which means that by definition the words are not in succession of each other, but rather in a specific order in the summary.

Each section in the ROUGE test gives three different scores, a Recall-, Precision- and F-score. The Recall-score is based on the amount of words that occur in reference summary also occur in candidate summary. The final score is calculated by the following equation 1:

$$\text{Recall} = \frac{\text{Overlapping words}}{\text{Total amount of words reference summary}} \quad (1)$$

The Precision-score is the exact opposite, where the score is based on the amount of words that occur in candidate summary also occur in reference summary. The score is calculated in by

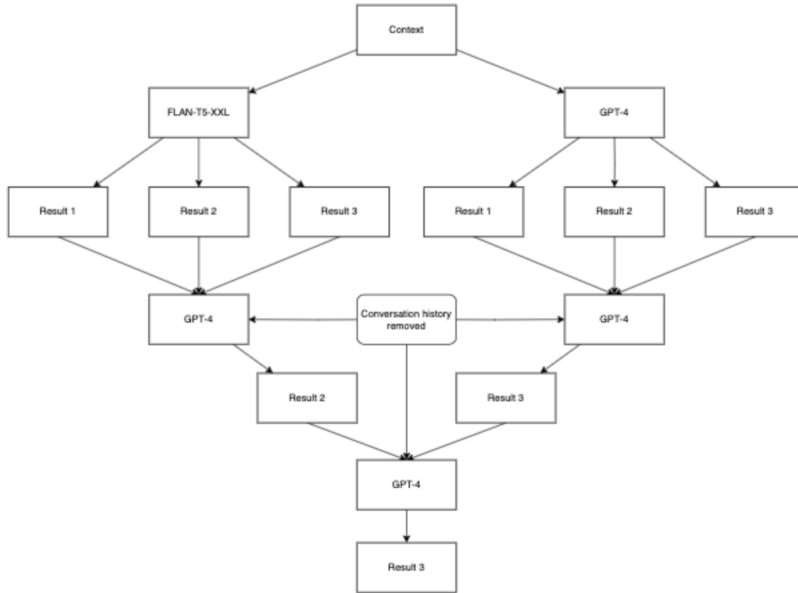


Figure 1: Best summary selection process

following equation 2:

$$\text{Precision} = \frac{\text{Overlapping words}}{\text{Total amount of words candidate summary}} \quad (2)$$

Finally, the F-score is the harmonic mean between Precision and Recall, as stated by Alvin (2022), given by the following equation 3: [2]

$$\text{F-score} = 2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \quad (3)$$

3.2.2 Best summary

As an alternative means to determine the better model out of the two mentioned above, GPT-4 is asked which summary is best. GPT-4 has no memory what it has written when the conversation is deleted, which gives a clear, neutral view to determine the best.

Consider the following example for the preliminary experiment: For each previously used contexts, the best F-score is chosen, with each having three results given by FLAN-T5-XXL and GPT-4. First, GPT-4 decides which result out of these three is best for FLAN-T5-XXL and for GPT-4. Next, GPT-4 is asked to decide the best summary out of the two 'best of' summaries. This leads to one summary created by either FLAN-T5-XXL or GPT-4 that is labeled as best for the context. This multi-step process is made visible in Figure 1 from top to bottom.

4 Experimental set up

From the start, the main question for Pega was, how to determine which model to use for each specific use case. This depends on different aspects as given in section 3.1.3. Therefore, at the start

of this paper, the following research question was formulated:

How could the summarization capabilities of various Large Language Models be compared on summarization tasks for both public- and domain specific data and models?

To answer this question it is necessary to perform multiple experiments where the summarization capabilities of Large Language Models are made visible. It is a useful aspect to be able to compare the current practise with external models to see if change is necessary or redundant.

This is done through three different experiments, starting with a preliminary experiment, followed by experiments on the News data set and the Pega documentation data set. The first experiment is to gain experience with the model outputs and the ROUGE-scores. Subsequently the other two experiments, for which GPT-4 is used to generate the ground truth to compare the other summaries with. In these experiments, public service GPT-3.5-turbo is compared to two open source Large Language Models, Pythia and MPT-7B, which are fine-tuned on Pega documentation.

4.1 Preliminary experiment

The preliminary experiment is carried out as a comparison of two models, FLAN-T5-XXL and GPT-4. Both models were given four documents to summarize, which were the exact same to see the difference in ability. The models were asked to summarize the context with nine different prompts, or input, which could show an influence of this. For each prompt, the models were asked to generate three summaries to see if there would be difference. These three summaries were used to compare to the reference and assign ROUGE-scores to.

The preliminary experiment was done by using SageMaker through Amazon Web Services (AWS), where a notebook with code was available to use the different versions of FLAN. [3] In this notebook it was possible to adjust the text and prompts, which gave the possibility to implement own input. Some prompts were already present in the notebook, to which some were added to give more insights to the influence of the prompts. To be able to give the scores to the summaries, some code was added to use the ROUGE scores. The references that were used, were human created to see how well the models would perform.

To compare the results of FLAN to another model, the chat function of GPT-4 in Microsoft Azure OpenAI Service was used. [15] This is a simple program that gives the user the possibility to use GPT-4 and ask it questions. In here the same context and prompts were used to ask the model to create summaries of the context. The summaries were then implemented in a PyCharm document to give the ROUGE scores to them. This way it was possible to compare the performance of the two models.

GPT-4 was also used to determine the best summary out of the two models as described above in section 3.2. GPT-4 was given two summaries, one of FLAN and one of GPT-4, and had to determine which was best.

4.2 News data set

The dataset that is used for this experiment, is a dataset with 280.000+ articles from CNN and DailyMail. [28] These articles can be used for extractive and abstractive summarization. Extractive summarization takes the most important sentences from a text and combines those to create a summary. Abstractive summarization creates new sentences and uses different words than the

actual text, which gives a new text. [31] In this experiment, the models got a sample of 500 articles from the dataset to summarize. These summaries are compared to summaries created by GPT-4 to be able to assign a ROUGE score to them. It is chosen not to use a public data set with human reference as for the Pega internal data there are no human written references available either. GPT-4 is chosen as it is the best performing model out there, with a relatively easy access.

The setup for the news data set experiment was somewhat similar, because the models are again asked to create summaries, but this time it was in a different environment. First up, GPT-3.5-turbo was used for the creation of summaries and it was done through a simple python script with an OpenAI linkup for API validation. The created summaries were again compared to the reference summaries to be able to give ROUGE scores. This time the reference summaries are not human created, as this would be too time consuming. The references are created by GPT-4 as this model is considered as one of the better models to exist, which caused it to be used as ground truth.

To compare the results of GPT-3.5-turbo to the current models, Pythia and MPT-7B are also asked the same task to be able to see how they perform. Both models were asked to create summaries of the articles in the news data set and compared to the reference summaries created by GPT-4. Both models are run through AWS and are callable with the right authorization. [3]

Some lines in the news data set gave errors when creating a summary, because the model could not handle the punctuation in the lines. These lines were stripped of interfering punctuation to provide the model with an acceptable line of text. Eventually these summaries were compared to the summaries of GPT-4 to assign a ROUGE score to them and compare them to the scores of GPT-3.5-turbo.

4.3 Pega documentation data set

The execution of the Pega documentation experiment did not differ much from the News data set. The only difference is the input text that needed to be summarized, which in this case was Pega related texts. The same models were used and tested in the same way, to see if the different contexts would be of any kind of influence and result in different scores. This experiment is most important to Pega, as this experiment involves actual data that will be summarized to ease the process of the employees.

In this experiment the three models got 248 Pega documents that needed to be summarized as well as possible. These are filtered to contain a minimum of 2500 characters and a maximum of 10000, to see how the models perform on large texts. Next, are the same steps as previously mentioned, the comparison to the GPT-4 summaries and assigning the ROUGE-scores.

The reason the models from the preliminary experiment were not used for the news data set and the Pega documentation data set, is that FLAN-T5-XXL was not capable of running this amount of data. The way it was setup at the time of testing resulted in multiple errors, which caused the switch to the other internal models. GPT-4 was not used in these experiments, because it is used as reference, as it is one of the better models to use.

4.4 News- and Pega documentation data set

To determine the best summaries for the News data set and Pega case experiments, a different setup is necessary. The models will create multiple unique summaries and only the summary with

FLAN-T5-XXL					
ID	Avg. Recall	Avg. Precision	Avg. F	Avg. length	St. Dev.
1	0.1982	0.3340	0.2300	101.5	55.7
2	0.7377	0.2365	0.3410	74.0	19.8
3	0.5125	0.1160	0.1860	58.0	9.3
4	0.1078	0.2136	0.1019	143.7	83.0
Total avg. Score	0.3890	0.2250	0.2147		

Table 2: The scores of FLAN-T5-XXL for the preliminary experiment

GPT-4					
ID	Avg. Recall	Avg. Precision	Avg. F	Avg. length	St. Dev.
1	0.2720	0.3816	0.3013	426.4	132.7
2	0.3864	0.3708	0.3611	288.3	94.6
3	0.2683	0.3067	0.2708	348.8	97.3
4	0.1701	0.2412	0.1828	355.2	188.1
Total avg. Score	0.27419	0.32507	0.27899		

Table 3: The scores of GPT-4 for the preliminary experiment

the highest F-score will be picked for further use as ground truth. The chance that two or three models create the best summary for the same text will be very small, which means each summary has to be rated individually. For each model the summary with the highest score F-score will be picked and the ID of that summary is collected. For the other two models the summary of that same ID will be collected to compare the three models. This is done by asking GPT-4 to give a score from 1 to 10 to each summary in relation to the original text. This will give an insight to the quality of the summary in addition to the ROUGE scores.

5 Results

To be able to answer the research question, multiple experiments have been carried out. Descriptions and results of the experiments are presented below.

5.1 Preliminary experiment

The first experiment that has been carried out is the preliminary experiment to see how these models work and perform and gain experience with the ROUGE evaluation method. By doing this preliminary experiment, an insight is given to the performance of the models for creating summaries with small texts or small context, which can be a difficult task to perform.

5.1.1 Results

It would be expected that GPT-4 would perform better in all aspects, because it outperforms other models in other benchmarks, and more importantly, it was the model to generate the ground truth reference summaries. This is for the preliminary experiment not the case for all points. The average

ID	Best summary	Model
1	"Effective 2/1/20, the insured moved..."	GPT-4
2	"The user reached out to U+Bank support..."	GPT-4
3	"The chat involves a customer and..."	GPT-4
4	"The email addresses the need for..."	GPT-4

Table 4: Best summaries and its model

Model	Avg. Recall	Avg. Precision	Avg. F
GPT-3.5-turbo	0.44665	0.60061	0.50668
Pythia (internal)	0.28345	0.16254	0.19216
MPT-7B (internal)	0.24151	0.47134	0.28590

Table 5: The scores of the models for the news data set

Recall is higher for FLAN-T5-XXL than for GPT-4. This is not a surprise, because the summaries that are created by FLAN-T5-XXL are much shorter. The reference summaries that the models were compared with were also short. This caused that most words from the reference summary are in the summary of FLAN-T5-XXL, which causes the ROUGE metrics to give it a high Recall score.

The other scores are higher for GPT-4 than for FLAN-T5-XXL, because the created summaries are much more elaborate, which is visible in Table 3. This results in a lower Recall-score, because GPT-4 uses extra words to summarize, which are not in line with the reference summaries. Moreover, GPT-4 has a higher Precision-score, because it has a higher chance to have words in common with the reference summary. Furthermore, the F-score is calculated with the Recall- and Precision-score, so when the scores are similar, the F-score will also be. However, when one score is high and the other is very low, the F-score will become a low score too, as visible for ID 3 in Table 2.

5.1.2 Best summary

For all used contexts, the best FLAN-T5-XXL summary was compared with the best GPT-4 summary. For each context, as visible in Table 4, the GPT-4 summary was the best, even though GPT-4 did not know it was its own summary. These summaries were more complete and consisted of the most important information. The prompt that lead to the better summaries was '[text] Can you generate a short summary of the above paragraph?'. The summaries of FLAN-T5-XXL were a lot shorter and less completed with the important information. The full summaries are available in Appendix A.

5.2 News data set

To be able to show the abilities for summarization of different models, a larger dataset with news articles is used for summarization.

5.2.1 Results

On average, the length of the original documents that had to be summarized, was 3984 characters long and GPT-4 was capable of slimming these texts down to 433 characters in its summaries.

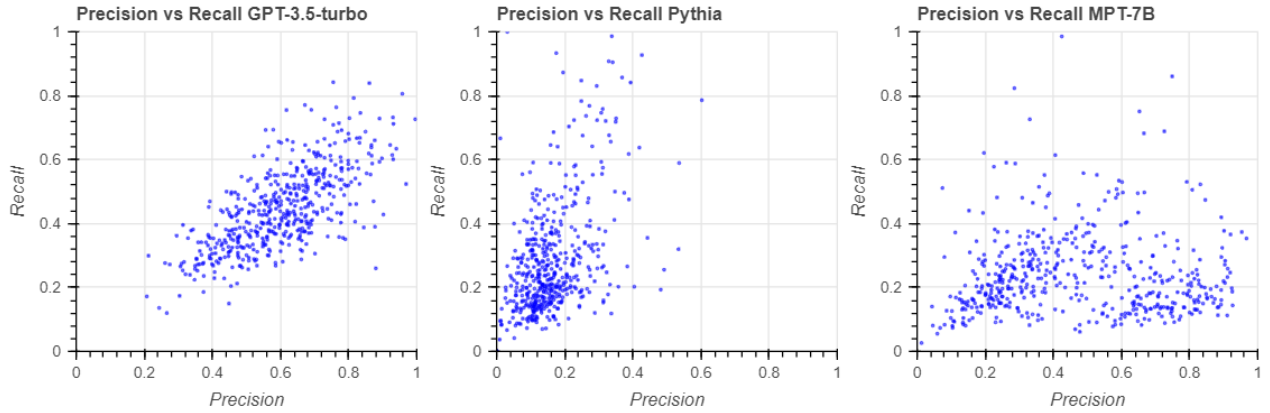


Figure 2: Precision and Recall for the news data set

Model	Mean	Median	St. Dev.
GPT-3.5-turbo	615.3	604.0	130.1
Pythia	272.4	227.0	185.6
MPT-7B	1235.7	782.0	1007.1

Table 6: Summary length statistics for model summaries

When looking at the results of the ROUGE scores, it is clear that GPT-3.5-turbo performs better at summarization than the two internal models Pega has. The Recall-, Precision- and F-score are all higher than the scores of the other models, as shown in Table 5. This division in scores was expected, as the two internal models have been fine tuned on Pega related data, which causes them to have a disadvantage and they are also considerably smaller than GPT-4.

Moreover, the news data set that is used is publicly available and a popularly used one. This means GPT-3.5-turbo might have been trained on this data set, which can be a reason for the higher scores. Additionally, the texts that are in this data set are similar to other text, which could contribute to these scores.

Precision and Recall are the scores that ROUGE is based on and the differences in the models are significant. As visible in Figure 2, GPT-3.5-turbo is the consistent scorer, with a distribution where the Precision and Recall increase almost identical. Pythia is a bad performer on Precision, as almost all cases have a Precision score below 0,4. When the Precision increases the Recall increases, but most dots appear at the bottom. MPT-7B gets low Recall scores, as it creates summaries that include information that the reference summary does not have. An increase in Precision does not contribute to a rise in Recall, as most dots are below 0.4 on Recall. Note that in contrast to other uses of Precision and Recall in machine learning there is no inverse relationship between Precision and Recall. This may be a function of the fact that the length of the summaries can vary.

When looking at the statistics in Table 6, large differences show between the three models. GPT-3.5-turbo is most consistent, with a the mean and median around the 600 character mark and standard deviation of 130.1 characters. Pythia creates much shorter summaries than the other two models, with a high standard deviation compared to the mean. MPT-7B is the most inconsistent model out of the three with a standard deviation of 1007.1, which is extremely high.

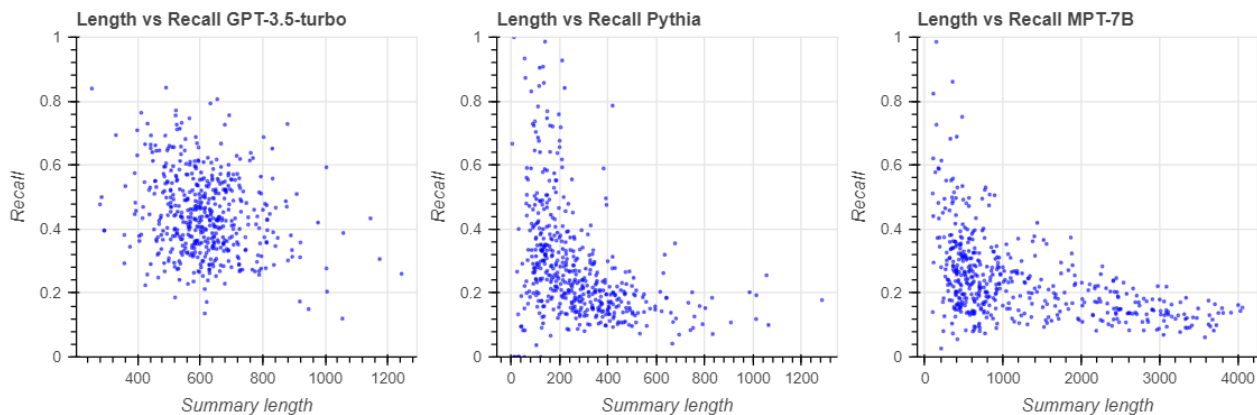


Figure 3: Summary length and Recall for the news data set

ID	Model	Summary	F-score	GPT-4 score
193	Pythia (internal)	"Yes. another suspected suicide bombing..."	0.68186	7
	GPT-3.5-turbo	"Two suspected suicide bombings occurred..."	0.5457	9
	MPT-7B (internal)	"Volgograd explosives attacks suspected to..."	0.47554	6
279	GPT-3.5-turbo	"Five Russian Sukhoi fighter jets have..."	0.87506	9
	Pythia (internal)	"The five Russian Sukhoi fighter jets are..."	0.21785	4
	MPT-7B (internal)	"The article discusses the arrival of five..."	0.53161	6
345	MPT-7B (internal)	"Bart Simpson appeared before the judge..."	0.80065	8
	GPT-3.5-turbo	"A man named Barton Simpson..."	0.51879	10
	Pythia (internal)	"Bart Simpson faced criminal charges for..."	0.18118	6

Table 7: Summary and score for each model

The connection between the length of the summary and the Recall scores are shown in Figure 3, where there are again large differences between the three models. The large difference is between GPT-3.5-turbo on the one hand and Pythia and MPT-7B on the other. GPT-3.5-turbo has a distribution where an increase in Summary length gives a decrease in Recall, but this decrease is very slim. The other two models' relationship can be described as inverse, as they decrease in Recall as the Summary length increases. This is because an increase in Summary length will only lead to an increase in less important information, which gives the lower Recall score.

5.2.2 Best summary

As stated in section 3.2.2, the setup for the news data set for the determination of the best summary is a bit different than the preliminary experiment. Each models' highest F-score is chosen and the ID of that summary. The other two models' summaries are also collected for the same ID. This way it is possible to compare the three models for the same subject, all shown in Figure 7. As visible, the score of GPT-4 is in favor of GPT-3.5-turbo for all instances, as it is most accurate and concise. However, this is not in line with the F-scores assigned to each summary. It is possible that the grading through GPT-4 is biased as it is based on the same algorithm as GPT-3.5-turbo.

The full summaries and the reasoning for the scores of GPT-4 are visible in Appendix B.

Model	Avg. Recall	Avg. Precision	Avg. F
GPT-3.5-turbo	0.45643	0.4668	0.45515
Pythia (internal)	0.26544	0.14118	0.1708
MPT-7B (internal)	0.31094	0.34387	0.3086

Table 8: The scores of the models for the Pega case

Model	Mean	Median	St. Dev.
GPT-3.5-turbo	634.7	616.0	167.9
Pythia	305.0	284.5	201.5
MPT-7B	846.8	701.0	565.1

Table 9: Statistics of model summaries for the Pega case

5.3 Pega documentation

To make this research better fitted for Pega, it is necessary to process their data to give recommendations on the present methods. Therefore the third experiment is about Pega data, which will be summarized and tested the same way as the news data set experiment. This is to show the possibilities on summarization with the current methods and the influence of different texts.

5.3.1 Results

The experiment with the Pega data gave different results for the ROUGE scores, but again GPT-3.5-turbo scored the highest in each section, as shown in Table 8. Compared to the results from the News data set test the models perform slightly worse on the Pega data. This is due to the Pega data being less widely used types of text and thus the text being structured in an unnatural way. For comparison, the news data set used news articles which were written for a general audience to read and understand. This caused the models to understand the texts less and unable to perform on the same level as the news data set test.

Figure 4 shows again the Precision and the Recall, this time for the Pega case. As stated above the results are a bit different than the News data set, which is clearly visible. The Precision scores of GPT-3.5-turbo are now even more aligned with the 45 degree linear line, where an increase in Precision will lead to a similar increase in Recall. Pythia scores are again in the bottom left corner of the plot and an even bigger proportion of the data is close to the 0.2 line of Precision than the news data set. The largest difference is visible for MPT-7B, where the distribution has moved to the left of and around the 0.4 line of Precision in stead of the distribution below the 0.4 line of Recall. This is due to the creation of shorter summaries by MPT-7b, which causes the Recall scores to be higher.

The Pega case has different context than the news data set, which results in different summaries. Table 9 shows the statistics of the summaries created by the three models. GPT-3.5-turbo and Pythia differ slightly with a small increase in all factors. They have to include more characters to be able to create a summary. MPT-7B on the other hand, shows an significant reduction in all factors, with the mean by 31.5% and the standard deviation by 43.9%.

As for the connection between Summary length and Recall, the division of GPT-3.5-turbo versus Pythia and MPT-7B is still present. The distribution of GPT-3.5-turbo is now more scattered than

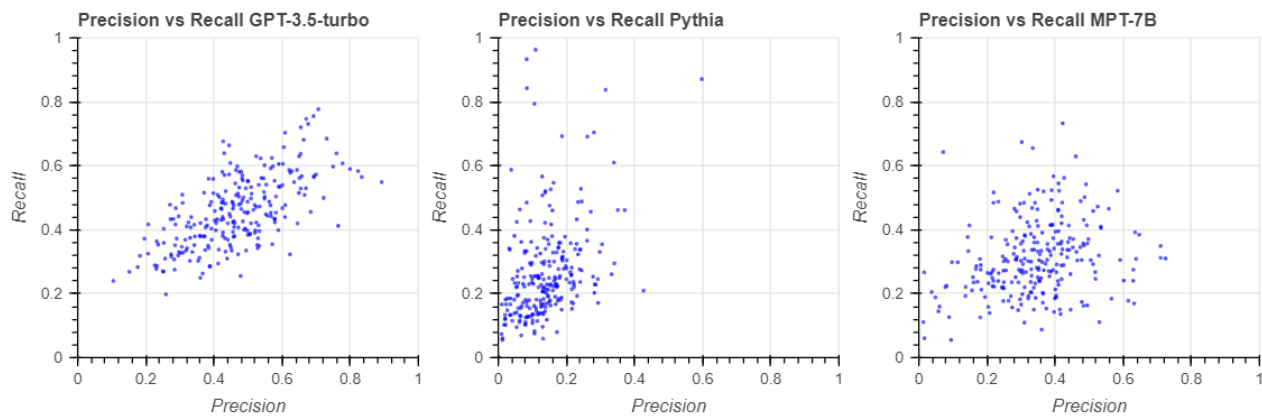


Figure 4: Precision and Recall for the Pega case

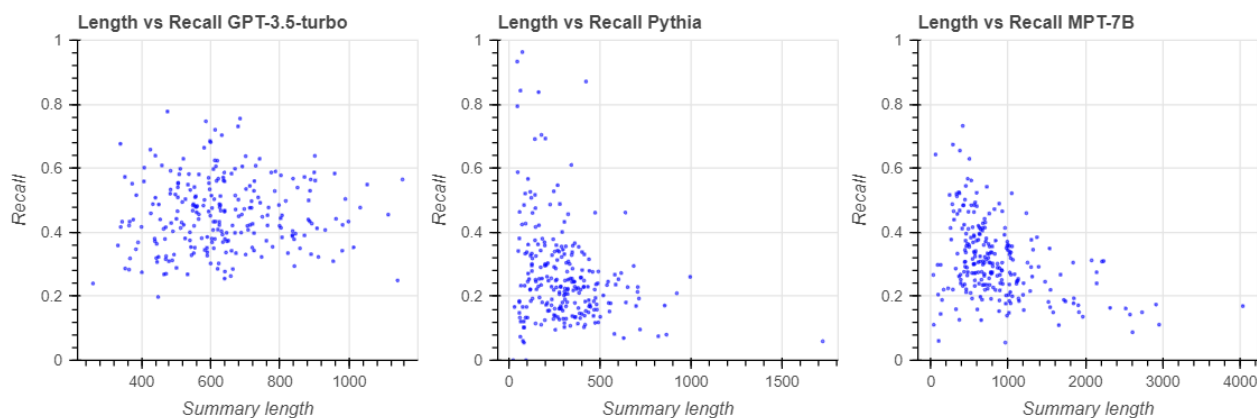


Figure 5: Summary length and Recall for the Pega case

at the news data set, with higher Summary length now leading to an even smaller decrease in Recall. Pythia and MPT-7B still show an inverse distribution on the Summary length and Recall scores. Pythia created a bit longer summaries, but not significantly to differ from the scatter plot from the news data set. MPT-7B created much shorter summaries and as a consequence of these shorter summaries, the Recall scores of MPT-7B are higher on average, as visible in Figure 5.

5.3.2 Best summary

The same structure was used to determine the best summary according to GPT-4. The F-score was used to be able to attain the ID, for which the other two models' summaries were gathered. GPT-4 was then asked to grade each summary, all made visible in Figure 10. As shown, GPT-4 again picks GPT-3.5-turbo for each instance as the best summary as they are most concise. This is again not in line with the given F-scores to the summaries. As stated in section 5.2.2 GPT-4 is based on the same algorithm as GPT-3.5-turbo which causes it to have the same reasoning and therefore rating the summaries the best.

The full summaries and the reasoning of GPT-4 are available in Appendix C.

ID	Model	Summary	F-score	GPT-4 score
2	MPT-7B (internal)	"Here's a summary of the above text: 1..."	0.5507	7
	GPT-3.5-turbo	"This text provides a set of instructions..."	0.28315	9
	Pythia (internal)	"The steps described above should..."	0.08185	5
33	Pythia (internal)	"To summarize the steps needed to add..."	0.70863	7
	GPT-3.5-turbo	"The text explains how to add a new..."	0.314	9
	MPT-7B (internal)	"The user can configure their application..."	0.39765	6
66	GPT-3.5-turbo	"The Pega Customer Decision Hub stores..."	0.74013	9
	Pythia (internal)	"The Pegadata.CDH.NegotiationOutcome..."	0.16218	5
	MPT-7B (internal)	"The negotiation outcomes are stored..."	0.39787	8

Table 10: Summary and score for each model

6 Discussion

This research looked at different LLMs and their ability to summarize text. This was done multiple times with different texts to see the performance of the LLMs. To determine the quality of the summaries that were created, a comparison was made to a preset ground truth created by GPT-4. The results indicate that the current internal models available to Pega do not perform sufficiently on summarization.

This research shows different ways to determine the best performing Large Language Model in summarization. By using the ROUGE-scores and the GPT-4 best summary, both quantitative- and qualitative analysis were used to show the results. The latter has the possibility to contain some form of bias, as the results visibly had a preference for the other model by OpenAI. This opens up the possibility for further research into GPT-4 to determine the presence of this bias.

6.1 Reflection on results

The results of this research show that some change is needed for Pega, should they want to use their internal models for summarization. The current models that are in use are outperformed by a non fine-tuned model that is publicly available. This is something that is not preferred must this use case become reality.

This outcome is not that big of a surprise, as the internal models were expected to perform worse. This is mostly because the internal models are fine-tuned on Pega data, which is not data for summarization. This data was used to train the models for question and answering and thus a different goal than the task set in this thesis. This is an issue that could not be resolved for this thesis, as the models that were used were already chosen by Pega. A solution to this problem could be to implement a different training for this model and give it data for summarization.

6.2 Limitations

As the data for the Pega case was internal and not publicly available, a limitation arises for further research on this subject. Therefore the generalizability of a part of the results is rather low, as the third experiment is not applicable for further research without access to Pega documentation.

However, the set up for the News data set was exactly the same and done with public data, so the set up is reusable, solely the used data is not.

Furthermore, the access to most of the models is limited due to payment. To use the GPT, Pythia and MPT-7B models access is required to AWS or another API, which will cost the user money depending on the amount it is used. For this thesis the access was arranged through Pega, which caused it to be without expenses for the author. This is not as easy for everyone to redo, as the costs for access to these models can be high. With the limited access to models due to payment, another limitation arises. It is possible for most models to use a free API that provides access to the model to anyone. These APIs are however limited to a certain amount of requests, so a certain amount of questions it can answer or summaries it can create. Additionally, the amount of text in can process is limited on the free API, to persuade people into buying the upgraded version. To be able to recreate the experiments done in this research, access to models through payment is necessary, which can be one of the bigger thresholds.

Another limitation of this research, is that in the News data set and the Pega documentation experiments, no prompt engineering was performed to determine the best prompt for summarization. However, this can be of influence to the performance of the models, as the models can do something different depending on the prompt. It could be that the results would have been different if another prompt was used to instruct the models. This was done by instructions of Pega, because they do not want to have a single prompt for each model in their systems. This lead to the usage of a single prompt for each model in the News data set and Pega documentation experiments.

6.3 Future work

To better understand the possibilities of Large Language Models, further research should focus on a wider selection of models to see the performance of those that are available. It should also look at the influence of fine-tuning on the models, as this could lead to different results. The internal models were fine-tuned on question and answering data and therefore led to worse results. Training these models on data for summarization would lead to better results, as the model would be trained to perform better on this specific task.

Furthermore, future research is necessary to be able to show the influence the length of the prompts that were used to create the summaries. It could be very well possible that the usage of different prompts would lead to different or better summaries.

Additionally, the influence of the texts that need summarization should be considered for further research as they play a key role in this research. Using less publicly available data could lead to different results, as the models may not be trained on this data. For example, GPT models are trained on data until 2021, which means data after that date is unknown to the model. If the model is asked to summarize this data, the results of the summary comparison could differ.

Another subject that could be of interest for further research, is the consistency of GPT-4 for summarization. The main question would be what if GPT-4 was asked again to create summaries on the given data, would they be the same. As in the preliminary experiment the models were asked to create three different summaries per prompt, which lead to difference in summary. This could also be the case for the other two experiments, which means the results could be higher, should the summary of GPT-4 not be the best it can create.

7 Conclusions

This research aimed to determine the abilities in summarization of multiple Large Language Models. This was done to be able to give Pega a better insight into their abilities today and possible changes. Tests were carried out with open data sets as well as company proprietary data, to be able to compare publicly available services against open source models fine-tuned on company data. Based on the quantitative and qualitative analysis done with different data sets, it can be concluded that the internal models that Pega has available are not capable to perform on par with the public services on summarization tasks. The results are useful for certain use cases, where use of publicly available services may not be cost effective or appropriate given the proprietary nature of the data.

References

- [1] AKGONUL, K. The future of business will be powered by Generative AI. <https://www.pega.com/insights/articles/future-business-will-be-powered-generative-ai>, March 2023. Last accessed 20 April 2023.
- [2] ALVIN, T. P. Introduction to text summarization with ROUGE scores. <https://towardsdatascience.com/introduction-to-text-summarization-with-rouge-scores-84140c64b471>, March 2022. Last accessed 26 April 2023.
- [3] AMAZON. Amazon SageMaker. <https://aws.amazon.com/sagemaker/>. Last accessed 09 June 2023.
- [4] BIDERMAN, S., SCHOELKOPF, H., ANTHONY, Q., BRADLEY, H., O'BRIEN, K., HALLAHAN, E., KHAN, M. A., PUROHIT, S., PRASHANTH, U. S., RAFF, E., SKOWRON, A., SUTAWIKA, L., AND VAN DER WAL, O. Pythia: A suite for analyzing large language models across training and scaling. <http://arxiv.org/abs/2304.01373>, 2023.
- [5] BODEN, M. A. *Artificial Intelligence*, 2nd ed. Academic Press, 1996.
- [6] CHUNG, H. W., HOU, L., LONGPRE, S., ZOPH, B., TAY, Y., FEDUS, W., LI, Y., WANG, X., DEGHANI, M., BRAHMA, S., WEBSON, A., GU, S. S., DAI, Z., SUZGUN, M., CHEN, X., CHOWDHERY, A., CASTRO-ROS, A., PELLAT, M., ROBINSON, K., VALTER, D., NARANG, S., MISHRA, G., YU, A., ZHAO, V., HUANG, Y., DAI, A., YU, H., PETROV, S., CHI, E. H., DEAN, J., DEVLIN, J., ROBERTS, A., ZHOU, D., LE, Q. V., AND WEI, J. Scaling instruction-finetuned language models. <http://arxiv.org/abs/2210.11416>, 2022.
- [7] COOPER, K. OpenAI GPT-3: Everything you need to know. <https://www.springboard.com/blog/data-science/machine-learning-gpt-3-open-ai/>, November 2021. Last accessed 20 April 2023.
- [8] ELEUTHERAI. Pythia: Interpreting transformers across time and scale. <https://github.com/EleutherAI/pythia>, 2023. Last accessed 20 June 2023.
- [9] ELEUTHERAI. Releases. <https://www.eleuther.ai/releases>, February 2023. Last accessed 20 June 2023.

- [10] FOUNDATION, A. S. Apache license version 2.0. <https://www.apache.org/licenses/LICENSE-2.0>, January 2004. Last accessed 10 May 2023.
- [11] GAO, L., BIDERMAN, S., BLACK, S., GOLDING, L., HOPPE, T., FOSTER, C., PHANG, J., HE, H., THITE, A., NABESHIMA, N., PRESSER, S., AND LEAHY, C. The Pile: An 800GB dataset of diverse text for language modeling. <http://arxiv.org/abs/2101.00027>, 2020.
- [12] KERNER, S. M. What is a large language model (LLM)? <https://www.techtarget.com/whatis/definition/large-language-model-LLM>. Last accessed 18 April 2023.
- [13] LI, C., AND XING, W. Natural language generation using deep learning to support MOOC learners. *International Journal of Artificial Intelligence in Education* 31 (2021), 186 – 214.
- [14] LIN, C.-Y. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out* (Barcelona, Spain, July 2004), Association for Computational Linguistics, pp. 74–81.
- [15] MICROSOFT. Azure OpenAI service. <https://azure.microsoft.com/nl-nl/products/cognitive-services/openai-service>. Last accessed 09 June 2023.
- [16] MOSAICML NLP TEAM, T. Introducing MPT-7B: A new standard for open-source, commercially usable LLMs. <https://www.mosaicml.com/blog/mpt-7b>, May 2023. Last accessed 27 June 2023.
- [17] NARANG, S., AND CHOWDHERY, A. Pathways language model (PaLM): Scaling to 540 billion parameters for breakthrough performance. <https://ai.googleblog.com/2022/04/pathways-language-model-palm-scaling-to.html>, April 2022. Last accessed 20 April 2023.
- [18] OPENAI. Models. <https://platform.openai.com/docs/models/overview>. Last accessed 12 June 2023.
- [19] OPENAI. OpenAI. <https://openai.com/>. Last accessed 18 April 2023.
- [20] OPENAI. Pricing. <https://openai.com/pricing>. Last accessed 10 July 2023.
- [21] OPENAI. GPT-4 technical report. <http://arxiv.org/abs/2303.08774>, 2023. Last accessed 10 July 2023.
- [22] OPENAI. OpenAI: GPT-4. <https://openai.com/research/gpt-4>, March 2023. Last accessed 12 June 2023.
- [23] PEGASYSTEMS. Office locations worldwide. <https://www.pega.com/about/office-locations>. Last accessed 19 April 2023.
- [24] PEGASYSTEMS. Pega: Low code platform for AI decisioning and workflow automation. <https://www.pega.com/>. Last accessed 18 April 2023.
- [25] PEGASYSTEMS. Pega software solutions. <https://www.pega.com/products>. Last accessed 19 April 2023.

- [26] PEGASYSTEMS. Pega announces Pega GenAI to infuse generative AI capabilities in pega infinity '23. <https://www.pega.com/about/news/press-releases/pega-announces-pega-genai-infuse-generative-ai-capabilities-pega-infinity>, May 2023. Last accessed 10 July 2023.
- [27] PEGASYSTEMS. Pega to integrate new generative AI technology to accelerate low-code app development and improve customer engagement. <https://www.pega.com/about/news/press-releases/pega-integrate-new-generative-ai-technology-accelerate-low-code-app>, March 2023. Last accessed 20 April 2023.
- [28] PENUGONDA, G. S. CNN-DailyMail news text summarization. <https://www.kaggle.com/datasets/gowrishankarp/newspaper-text-summarization-cnn-dailymail>, n.d. Last accessed 09 June 2023.
- [29] RAFFEL, C., SHAZEER, N., ROBERTS, A., LEE, K., NARANG, S., MATENA, M., ZHOU, Y., LI, W., AND LIU, P. J. Exploring the limits of transfer learning with a unified text-to-text transformer. *CoRR abs/1910.10683* (2019).
- [30] ROBERTS, A., AND RAFFEL, C. Exploring transfer learning with T5: the Text-To-Text Transfer Transformer. <https://ai.googleblog.com/2020/02/exploring-transfer-learning-with-t5.html>, February 2020. Last accessed 28 April 2023.
- [31] ROY, A. Understanding automatic text summarization-1: Extractive methods. <https://towardsdatascience.com/understanding-automatic-text-summarization-1-extractive-methods-8eb512b21ecc>, August 2020. Last accessed 09 June 2023.
- [32] TAORI, R., GULRAJANI, I., ZHANG, T., DUBOIS, Y., LI, X., GUESTRIN, C., LIANG, P., AND HASHIMOTO, T. B. Alpaca: A strong, replicable instruction-following model. <https://crfm.stanford.edu/2023/03/13/alpaca.html>, March 2023. Last accessed 10 May 2023.
- [33] TOUVRON, H., LAVRIL, T., IZACARD, G., MARTINET, X., LACHAUX, M.-A., LACROIX, T., ROZIÈRE, B., GOYAL, N., HAMBRO, E., AZHAR, F., RODRIGUEZ, A., JOULIN, A., GRAVE, E., AND LAMPLE, G. LLaMA: Open and efficient foundation language models. <http://arxiv.org/abs/2302.13971>, 2023.
- [34] TRUEFOUNDRY. Large language models for commercial use. <https://blog.truefoundry.com/all-about-license-for-llm-models/>, April 2023. Last accessed 10 May 2023.
- [35] ZHANG, H., LIU, X., AND ZHANG, J. Extractive summarization via ChatGPT for faithful summary generation. <https://arxiv.org/abs/2304.04193>, 2023.
- [36] ZHAO, T. Z., WALLACE, E., FENG, S., KLEIN, D., AND SINGH, S. Calibrate before use: Improving few-shot performance of language models. <http://arxiv.org/abs/2102.09690>, 2021.

Appendix

A

Preliminary experiment - Best summary

FLAN-T5-XXL results:

1. Please make the following changes - Effective 2/1/20, the insured has moved locations and forgot to tell me. Please change billing, mailing, location address to: 844 N. Broad St, Philadelphia, PA 19130. Built 1920 J
2. User wants to change the Billing Date. DMSUser4 has created a service request with Id : REQ-1234.
3. Customer wants to know about Family safeguards and controls.
4. I believe we need to find a path forward that is true to the definition of the word “hybrid” where we take advantage of working from home at times but still allow for ample time in the office, ultimately to drive innovation

GPT-4 results:

1. Effective 2/1/20, the insured moved to 844 N. Broad St, Philadelphia, PA 19130 and requests address updates for billing, mailing, and location. The building, built in 1920, has 3,000 sq ft, 3 stories, no sprinklers, and was updated in 2000. Contents coverage should remain at \$11,300 with no operational changes. Notarized documents have been uploaded to the portal, and a \$50 fee paid. Awaiting confirmation of the changes.
2. The user reached out to U+Bank support agent DMSUser4 for help with changing their billing date to the 21st of every month. DMSUser4 created a service request (REQ-1234) to address the issue, which can be tracked on the provided website. The user thanked DMSUser4 for their assistance.
3. The chat involves a customer and a Customer Service Representative (CSR) discussing Family safeguards and controls for the customer’s data plan after adding their son. The CSR explains the service, which monitors the son’s activities and sets restrictions, costs \$5 per month. The customer enrolls in the service, expresses satisfaction with the CSR’s assistance, and is invited to participate in an online survey for a chance to win a \$50 gift certificate.
4. The email addresses the need for a hybrid work approach in the FlexWork program, emphasizing the benefits of both remote work and in-office collaboration. The sender recommends employees within commutable distance to spend 2-3 days per week in the office to foster innovation, teamwork, and personal growth.

B

News data set - best summary

Summaries:

1.
 - GPT-3.5-turbo: Five Russian Sukhoi fighter jets have arrived in Iraq, with 25 more expected to be delivered under a contract between Moscow and Baghdad. The Iraqi Ministry of Defense said the jets will increase the combat capability of the air force and the rest of the armed forces in eliminating terrorism. The Su-25 has been in service for more than three decades and was part of the Iraqi air force under Saddam Hussein's regime. The announcement follows a comment by Prime Minister Nuri al-Maliki that recent militant advances might have been avoided if Iraq had proper air power, in the form of fighter jets that Iraq has been trying to get from the United States.
 - Pythia: The five Russian Sukhoi fighter jets are scheduled to arrive in Iraq on June 20th. They're the first of twenty-five aircraft ordered by the Iraqi government.
 - MPT-7B: The article discusses the arrival of five Russian Sukhoi fighter jets in Iraq under a contract between Moscow and Baghdad. The jets were being pulled toward hangars and were described as being in "good condition". The Su-25 has been in service for more than three decades and was part of the Iraqi air force under Saddam Hussein's regime. The announcement follows a comment by Prime Minister Nuri al-Maliki that recent militant advances might have been avoided if Iraq had proper air power, in the form of fighter jets that Iraq has been trying to get from the United States. "I'll be frank and say that we were deluded when we signed the contract" with the United States, al-Maliki told the BBC in the interview last week, which was released Friday. Iraq has now turned to Russia and Belarus to buy fighter jets, he said. "God willing, within one week, this force will be effective and will destroy the terrorists' dens," he said. U.S. officials were quick to reject al-Maliki's complaints. U.S. fighter jets have not been slow in coming, Pentagon spokesman Rear Adm. John Kirby said. The first two promised F-16s "weren't expected to be delivered until the fall, which is still months away", Kirby said. "And we were in the process of working towards that delivery". The advance of an al Qaeda splinter group "couldn't have been stemmed through the use of two particular fighter planes", he said.

2.
 - Pythia: Yes. another suspected suicide bombing struck the southern Russian city of Volgograd on Monday. killing at least 14 people and further highlighting Russia's security challenges as it prepares to host the Winter Olympics in less than six weeks. The explosion hit a trolleybus near a busy market during the morning rush hour. a day after a blast at Volgograd's main train station killed 17 people and wounded at least 35.
 - GPT-3.5-turbo: Two suspected suicide bombings occurred in the southern Russian city of Volgograd, killing at least 14 people and injuring dozens more. The attacks highlight Russia's security challenges as it prepares to host the Winter Olympics in less than six weeks. No one has claimed responsibility for the blasts, but they occurred several months after the leader of a Chechen separatist group pledged violence to disrupt the 2014 Sochi Winter Olympics that begin on February 7. The attacks are believed to be part of an attempt by terrorists to open a domestic front, sow panic and chaos, and trigger religious strife and conflicts in Russian society.
 - MPT-7B: Volgograd explosives attacks suspected to be terrorist sabotageAnother suspected suicide bombing struck the southern Russian city of Volgograd on Monday, killing at least 14 people and highlighting Russia's security challenges as it prepares to host

the Winter Olympics in less than six weeks. The explosion hit a trolleybus near a busy market during the morning rush hour, a day after a blast at Volgograd's main train station killed 17 people and wounded at least 35. Vladimir Markin, a spokesman for the country's federal investigation agency, told the state-run news agency RIA Novosti that both explosions were terrorist attacks. "This strike, which was cynically planned for the period of preparations for New Year's celebrations, is one more attempt by terrorists to open a domestic front, sow panic and chaos, and trigger religious strife and conflicts in Russian society," said a statement Monday by Russia's Foreign Affairs Ministry. "We will not back down and will continue our tough and consistent offensive" against terrorists, the ministry's statement said, adding that such an enemy "can only be stopped by joint efforts" involving the international community. The approaching Olympics have sparked concerns among security experts despite extensive security measures implemented by the Russian government. No one claimed responsibility for the Volgograd blasts, but they occurred several months after the leader of a Chechen separatist group pledged violence to disrupt the 2014 Sochi Winter Olympics that begin on February 7. The U.S. State Department considers the Caucasus Emirate a foreign terrorist group and has authorized a reward of up to \$5 million for information leading to the location of Umarov. The State Department also warned of impending threats against aviation targets and public places in Russia following the Boston Marathon bombings in 2013.

3.
 - MPT-7B: Bart Simpson appeared before the judge, Mr. Recorder Burns, to face charges of possession of a firearm and possession of a dangerous article on board an aircraft. He pleaded guilty to the latter charge and was ordered to do 100 hours of unpaid work in the community and to pay a fine of 800 pounds (\$1,200) to the court. The gun was seized and destroyed.
 - GPT-3/5-turbo: A man named Barton Simpson, nicknamed Bart, appeared in court in England and faced charges of possessing a firearm and a dangerous article on an aircraft. He pleaded guilty to the latter charge and was ordered to do community service and pay a fine. The article mentions the coincidence of his name being the same as a character from the TV show "The Simpsons."
 - Pythia: Bart Simpson faced criminal charges for possessing a firearm and carrying a dangerous weapon aboard an airplane. He pled guilty to both charges and received a sentence that included paying fines totaling \$1200 USD.

GPT-4 score reasoning:

1. Summary 1: 9/10 - It is concise, covers the main points, and provides context about the situation.
Summary 2: 4/10 - It is too short and lacks important details about the purpose of the jets, the historical context, and the Prime Minister's comments.
Summary 3: 6/10 - While it includes more details, it is too long and repetitive, making it less effective as a summary for readers who want a quick understanding of the situation.
2. Summary 1: 7/10 - It covers the basic information about the bombings and their impact on Russia's security challenges before the Winter Olympics. However, it does not mention the possible connection to a Chechen separatist group.

Summary 2: 9/10 - This summary effectively captures the key information from the text in a concise manner, including the bombings, casualties, security challenges, and the possible connection to a Chechen separatist group. It provides a clear and comprehensive overview of the main points.

Summary 3: 6/10 - While this summary includes more details about the bombings and the potential connection to a Chechen separatist group, it is longer and less concise than the other summaries. It also contains some repetitive information, making it less efficient as a summary.

3. Summary 1: 8/10 - This summary includes most of the important details, but it does not mention the coincidence of the names being the same as characters from "The Simpsons."

Summary 2: 10/10 - This summary accurately and concisely captures all the main points of the original text, including the charges, the guilty plea, the consequences, and the interesting coincidence of the names.

Summary 3: 6/10 - This summary is less accurate, as it states that Bart pled guilty to both charges, which is incorrect. Additionally, it does not mention the coincidence of the names being the same as characters from "The Simpsons."

C

Pega case - best summary

Summaries:

- GPT-3.5-turbo: The Pega Customer Decision Hub stores negotiation outcomes and related interaction history records in PegaDATA database tables. The `pegadata.cdh_data_negotiation_outcome` table contains properties such as agent ID, budget used, commissions, customer ID, and outcome. The `pegadata.cdh_negotiation_ih_join` table contains properties that join the negotiation outcome table to the interaction history file storage. These tables can be used to create business intelligence reports.
 - Pythia: The `Pegadata.CDH_NegotiationOutcome` table includes properties that summarize the results from negotiations between agents and customers. These properties include the agent's budget usage, commissions earned, and other details regarding the negotiation process.
 - MPT-7B: The negotiation outcomes are stored in the `'pegadata.cdh_data_negotiation_outcome'` and `'pegadata.cdh_negotiation_ih_join'` tables in the Pega Database. The former contains basic properties that describe the negotiation outcome while the latter joins the `pegadata.cdh_data_negotiation_outcome` table to the interaction history file storage.
- Pythia: To summarize the steps needed to add a new email provider to Pega Customer Decision Hub, follow these steps: 1) Create a new class with the parent class as `PegaMKT-Data-EmailServiceProvider`. 2) Configure the webhook URL. 3) Add the required extension data transforms: `SetupCustomHeader`, `TransformHeader`, and `NormalizeResponseData`. 4) Define a service to process the email delivery and bounce events from the email provider.

- GPT-3.5-turbo: The text explains how to add a new email provider in Pega Customer Decision Hub. It is recommended to add a new email provider for business-critical email accounts. The process involves creating a new class, configuring the webhook URL, setting up custom headers, defining a service to process email delivery and bounce events, and recording bounced emails in the customer’s profile. The built-in Custom provider type is available for easy configuration of less critical email accounts but does not allow custom headers and does not record bounce and delivery events in interaction history.
 - MPT-7B: The user can configure their application to add an email provider that is not included in the default list by creating a new class with the parent class as PegaMKT-Data-EmailServiceProvider. They can also configure the required extension data transforms, such as TransformHeader, SetupCustomHeader, and NormalizeResponseData. The service must invoke the NormalizeResponseData data transform, as well as the CaptureBounceInfo activity, to record bounced emails in the customer’s profile.
3. • MPT-7B: Here’s a summary of the above text: 1. Disable production traffic that requires up-to-date interaction history summary data. 2. Dematerialize all existing interaction history summary data sets. 3. For all interaction history summary data sets, click ”Manage” > ”Recreate aggregates”. 4. Ensure the interactionHistory/writeToIH setting in the Pega-DecisionEngine ruleset is set to false. 5. Turn off saving the last interaction history record time to Cassandra by setting the prconfig/ih/aggregates/last_customer_update_method/default setting to NONE. 6. Delete the Aggregates_Interaction History pre-aggregation data flow run. 7. Launch the MigrateInteractionHistoryToStream activity in the Rule-Decision-CPD ruleset. 8. Restart the system by performing a full cluster rolling restart. 9. Materialize interaction history summary data sets. 10. Wait for the aggregation run to complete and ensure that the interaction history summary pre-aggregation run starts. 11. When the interaction history summaries are up to date, enable production traffic.
- GPT-3/5-turbo: This text provides a set of instructions for recreating interaction history summaries after migrating to the InteractionFiles data set. The process involves disabling production traffic, dematerializing existing interaction history summary data sets, recreating aggregates, ensuring certain settings are in place, deleting pre-aggregation data flow runs, launching an activity, restarting the system, materializing interaction history summary data sets, and waiting for the aggregation run to complete before enabling production traffic again.
 - Pythia: The steps described above should help you migrate your interaction history summaries from the old format (the one stored in the relational database) to the new format (stored as files in the file repository). You can then start using them again with the latest version of the Decision Strategy rule form.

GPT-4 score reasoning:

1. The Pega Customer Decision Hub stores negotiation outcomes and related interaction history records in PegaDATA database tables. The pegadata.cdhd_data_negotiation_outcome table contains properties such as agent ID, budget used, commissions, customer ID, and outcome. The pegadata.cdhd_negotiation_ih_join table contains properties that join the negotiation outcome table to the interaction history file storage. These tables can be used to create

business intelligence reports. - Rating: 9/10

The Pegadata.CDH.NegotiationOutcome table includes properties that summarize the results from negotiations between agents and customers. These properties include the agent's budget usage, commissions earned, and other details regarding the negotiation process. - Rating: 5/10

The negotiation outcomes are stored in the pegadata.cdh_data_negotiation_outcome and pegadata.cdh_negotiation_ih_join tables in the Pega Database. The former contains basic properties that describe the negotiation outcome while the latter joins the pegadata.cdh_data_negotiation_outcome table to the interaction history file storage. - Rating: 8/10

2. Summary 1: Rating - 7 This summary provides a clear step-by-step process for adding a new email provider to Pega Customer Decision Hub. However, it lacks context about the importance of adding a new email provider for business-critical accounts and the limitations of the built-in Custom provider type.

Summary 2: Rating - 9 This summary is comprehensive and concise, covering the key points from the original text, including the recommendation for business-critical email accounts, the limitations of the built-in Custom provider type, and the steps involved in the process. It is easy to understand and provides a well-rounded overview.

Summary 3: Rating - 6 This summary includes some essential details about configuring the application to add a new email provider and the required extension data transforms. However, it lacks clarity and omits important information about the recommendation for business-critical email accounts and the limitations of the built-in Custom provider type.

3. Summary number 1: 7/10 - It provides a detailed step-by-step breakdown of the process, but it might be too lengthy for someone looking for a quick overview.

Summary number 2: 9/10 - It offers a concise and comprehensive overview of the process, making it easy to understand the main steps involved without going into excessive detail.

Summary number 3: 5/10 - It is too brief and lacks context, making it difficult for the reader to grasp the full scope of the process.