Engineering Department

Master's Degree Artificial Intelligence and Data Engineering

Università degli studi di Pisa
Accademic year - 2022-2023

PaperAI: Multi-article summarization app based on GPT3.5, BART and Sentence Transformers

STUDENTS:
Giulio Fischietti, Luca Di Giacomo, Giovanni Paolini

Business and Project Management

Contents

1 Introduction

In recent years, there has been a substantial surge in the publication of scientific papers. This remarkable growth can be primarily attributed to the emergence of new technologies, which have considerably facilitated information access for researchers, coupled with a higher level of collaboration among scholars from different fields. The advent of these novel technologies, ranging from sophisticated machine learning algorithms to cutting-edge artificial intelligence frameworks and complex computer programs, has effectively loosened the boundaries between previously unrelated sectors.

As an illustrative example, consider the healthcare sector. This sector has undergone a profound metamorphosis, embracing an array of technological advancements that have revolutionized it at its core. The integration of machine learning algorithms has enabled rapid diagnosis and enhanced patient care, with AI-powered systems demonstrating remarkable accuracy in disease detection. Indeed, the healthcare sector now stands as a perfect example of the symbiotic relationship between technology and research collaboration.

These new links which arose from the interconnection of different fields generated a vast amount of new papers that now included not only information from a single area of study but from many. There are so many papers now making it hard for researchers to keep up with all of them. The old way of reading these papers, by looking at the abstract at the beginning and reading the whole paper, takes too much time and the specificity of certain papers may lead scholars without expertise in a field into confusion.

Because of this, there is a need for a new way to retrieve papers. Researchers need a tool that can quickly go through lots of research articles and give short summaries. This is where artificial intelligence comes in. AI can help create a search engine that can not only find publications but also make short summaries of them. It's like having a smart assistant that helps researchers understand many papers all at once.

At its core, our tool harnesses the capabilities of AI-generated summaries to condense both the abstract and main body of scientific papers. These summaries act as entry points, enabling researchers to quickly grasp the fundamental contributions of each paper. Furthermore, our tool goes beyond traditional summarization by introducing an interactive query component. Researchers can input queries, which can be used along with the generated summaries to receive contextually relevant responses.

This dual-functionality approach offers several advantages. First, it saves researchers valuable time by providing an efficient means of surveying a wide range of papers. Instead of reading each paper in its entirety, researchers can now access summarized information, enabling them to identify relevant works swiftly. Secondly, the interactive query component bridges the gap between traditional static search engines and dynamic information exchange. Researchers can articulate their specific queries, enabling the tool to not only provide relevant summaries but also contextualize these summaries within the broader research landscape.

In this report, we will introduce this innovative scientific paper search engine and its AI-powered summarization capabilities. We will analyze its structure and the inner working of the AI models used.

2 Application design

To address the needs described in the previous section we designed an application that has different components: a web application and different services.
The web application is what will be used by the end user (researcher), while the services update and provide content to the application by scraping scientific articles, generating and indexing summaries that then will be saved into the database.
Below is shown a high level representation of the app.

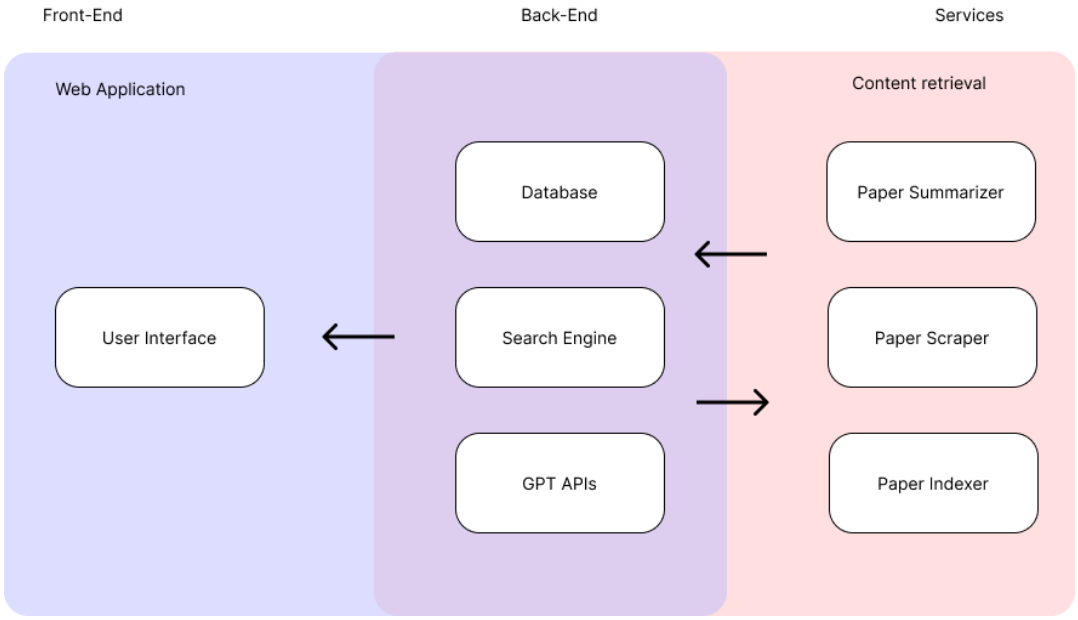

Figure 1: High level Application architecture

2.1 Web application design

The web application as previously mentioned is what will be used by the researcher, and it's the application that directly addresses its needs. It includes as can be seen from the previous schema, all the components in the front-end and back-end:

- user interface
- search engine
- database
- GPT3.5 APIs

2.1.1 Flow Diagram

The flow of the application is as already implemented in numerous business cases and it consists in an integration of a custom search engine with GPT APIs, here's how it works in detail:

1. The user types a query (question)
2. The query is converted by the sentence transformer into a vector
3. The search engine uses that vector as a query and returns the IDs of scientific papers most relevant to that query
4. The IDs are used to retrieve the summaries of those (relevant) scientific papers from our DB
5. These summaries and the user initial query are used as input to GPT APIs, being able to answer the user question by using the context given by the summaries of papers
6. The answer is then returned back to the user interface in the form of a single article

A more detailed view of the web application design is shown below:

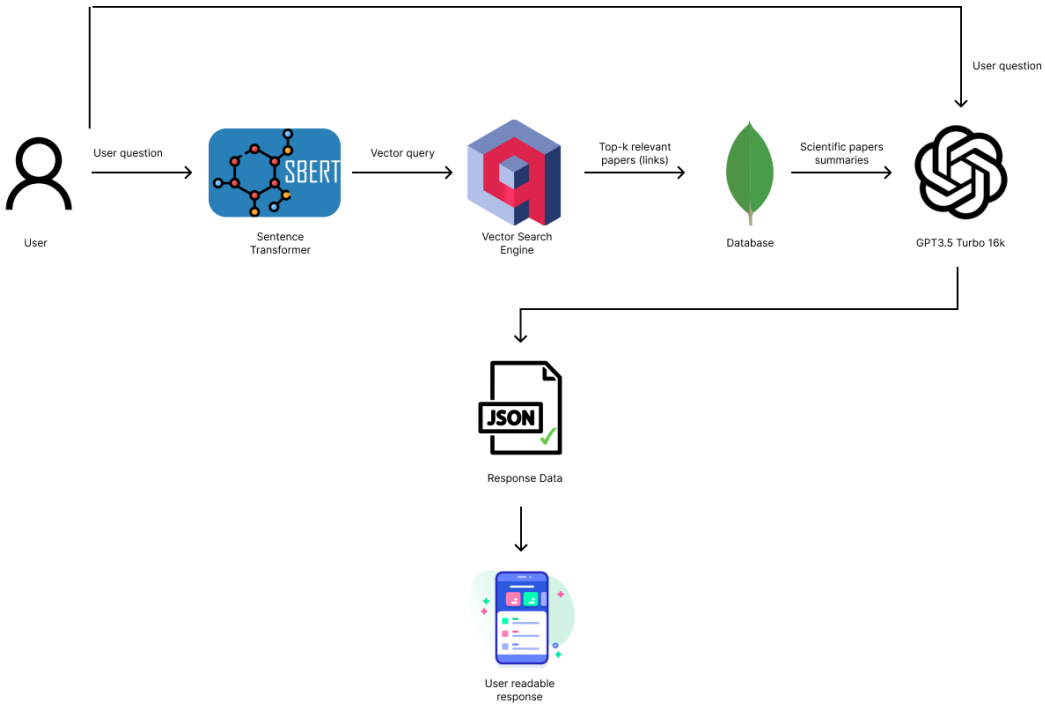

Figure 2: Web application architecture

2.2 Services

In order to keep updated content in our application, we designed three services

- Paper scraper
- Paper summarizer
- Paper indexer

2.2.1 Paper scraper

The paper scraper is a simple web scraping script, that when launched, retrieves the full body text of scientific articles from science direct with also other informations, such as:

- authors
- abstract
- publication date

All the informations are stored in the DB instance sincronously.

2.2.2 Paper summarizer

The paper summarizer retrieves all the papers on the database that are still not summarized and sequentially generates a summary with BART, updating the related document on the database with the appropriate field. More details about the summarization strategy will be discussed in the following paragraphs.

2.2.3 Paper indexer

The paper indexer is a script that indexes on Qdrant APIs the papers based on their summaries/abstracts, rather than on full body text: with this approach, we are not only able to save a lot of space from the search engine APIs, but also we avoid difficulties linked to the small maximum input length of the sentence transformers: full body texts, which can be composed of more than 10 thousand tokens, are too long to be vectorized by our (and most) sentence transformers that can take as max 512 tokens.
It is important to notice that despite running this script after the final sraping of the papers for simplicity purposes, this service can be easily implemented as a standalone asyncronous service that indexes the papers right after their scraping or immediately after the summarization process.

3 Summarization

Given the limited input size of chat GPT, consisting of 16.000 tokens, the need for a condensed version of papers emerged. This was a crucial step in the project since using only a part of the papers would result in the loss of context of the research article's topic. In order to achieve this, a summarization method was needed. To this day, there are many techniques for text summarization in use. The main methods are Extractive and Abstractive Summarization.

Extractive Summarization: The extractive approach involves selecting sentences or phrases directly from the source text to create a summary. Algorithms assess the significance of individual sentences based on features such as term frequency, position, sentence length, and other metrics. This method simplifies the process by utilizing existing content, yet potential challenges lie in maintaining coherence and capturing broader context. Here is an example:

Source text:

> ***Joseph and Mary*** *rode on a donkey to* ***attend*** *the annual* ***event*** *in* ***Jerusalem****. In the city,* ***Mary*** *gave* ***birth*** *to a child named* ***Jesus****.*

Extractive summary:

> *Joseph and Mary attend event Jerusalem. Mary birth Jesus.*

As can be seen above, the words in bold have been extracted and joined to create a summary.

Abstractive Summarization: The abstractive approach takes a more advanced route, generating summaries that go beyond sentence extraction. This technique requires a deeper understanding of the text's meaning and employs natural language processing (NLP) methods. With NLP we can understand the subtle meaning and relationship of words in a language, with this deeper understanding a Deep Learning model can be trained to perform a summarization that truly understands the essence of the text and does not only repeat words and sentences like in Extractive summarization. This method allows us to extrapolate a reliable summary from a paper while not losing track of the context or important parts of the research article. Here is an example:

Source text:

> *Joseph and Mary rode on a donkey to attend the annual event in Jerusalem. In the city, Mary gave birth to a child named Jesus.*

Abstractive summary:

> *Joseph and Mary came to Jerusalem where Jesus was born.*

The summary produced reflects more the original meaning of the source text and it looks more natural in structure.

3.1 Models and techniques used

A different number of summarization models were tested during the development phase of the project:

- MSMarco
- Pegasus
- BART

The main issue with most models was that, given the limited input size, the original text was split into chunks and fed to the model, but with this splitting they couldn't retain all the context of the papers, resulting in summaries that excluded important parts of the article. Specifically in the case of Pegasus by Google, the weight of the model, the issues with the setup of the model (missing libraries), and the limitations of the free version, ended up with the decision to discard this model completely.

Amongst the number of text generative models used, an extractive method was tried in conjunction with generative models. By using the term frequency as weight, a score was given to each sentence of the text after a step of text cleaning and normalization, then the list of sentences was sorted in descending order, and sentences from the top were picked until the number of maximum tokens in input for the model was reached. The produced text contained, in theory, the most important parts of the paper. This was a way to circumvent the input limit of tokens in most of the best-performing summarization models, but the generated summaries were not good still. In the end, BART was used as it was the method which performed better amongst the others.

BART(Bidirectional and Auto-Regressive Transformers) is an advanced language generation model. It's built on the transformer architecture, which utilized one or more self-attention layers to capture relationships between preceding and following words, allowing to keep track of the context of a word in the whole sentence or paragraph. This is necessary in the task of summarization and this model produced summaries of excellent quality during the tests that we performed. Fine-tuning could also be performed in order to improve the quality of generated text over a specific type of document, but this was not performed in the project since the dataset was not enough expansive (due to the limitations of scraping) to allow a good understanding of the data.

At first, for each paper, both the abstract and the body were used in order to generate a summary. This idea emerged from the need of maintaining the context of the paper. By using the abstract along the body chunks, each chunk should have maintained at least a portion of the whole context of the research article. This intuition gave some decent results but it was not producing summaries of higher quality than just using the bodies, on the contrary, the summaries occasionally included parts of text not inherent to the body chunk presented to the summarizer. In the end, the choice was to only use the bodies of papers for summarization.

3.2 Evaluation Metrics

In order to evaluate the quality of produced summaries, there is the need of some evaluation measure. Co-Selection measures such as precision and recall fall short when it comes to recognize that two sentences may convey similar information despite being phrased differently. For instance, given the two sentences:

> The visit of the president of the Czech Republic to Slovakia

> The Czech president visited Slovakia

These sentences encapsulate the same meaning and should be considered a match in a retrieval system, but considering only co-selection metrics they would not match. Content based similarity and task based similarity are more suitable metrics for this task.

3.2.1 Content-Based Measures

Cosine Similarity: One fundamental content-based metric is Cosine Similarity. It evaluates the similarity between a system summary (X) and a reference document (Y) by representing them in the vector space model. This metric measures how closely the vectors align, considering both direction and magnitude.

Unit Overlap: Another approach is Unit Overlap, which assesses the overlap between sets of words or lemmas in X and Y. It quantifies shared elements relative to the total elements in both sets, providing insights into the degree of commonality.

Longest Common Subsequence (LCS): LCS measures sequences of words or lemmas. It quantifies the length of the longest common subsequence between X and Y, factoring in their overall lengths and edit distance. The longer the common sequence, the stronger the semantic alignment.

N-gram Co-occurrence Statistics – ROUGE: In recent text summarization evaluations, ROUGE (Recall-Oriented Understudy for Gisting Evaluation) has gained prominence. This family of measures, based on the similarity of n-grams, was introduced in 2003. ROUGE-n computes the ratio of matching n-grams in candidate summaries to reference summaries. It offers a comprehensive evaluation of semantic alignment, allowing for a nuanced assessment.

3.2.2 Task-based Measures

Task-based evaluation methods do not analyze sentences in the summary. They try to measure the prospect of using summaries for a certain task. Various approaches to task-based summarization evaluation can be found in literature. We mention the three most important tasks – document categorization, information retrieval and question answering

4 Sentence Transformers

In order to use Artificial Intelligence to search among the collection of documents, we generated for each document a context-aware embedding using Sentence Transformers. In the offline process of our search engine, we concatenated the abstract to the title, then we used Allenai-specter, which is a sentence transformer specifically trained to produce embeddings out of scientific papers to get an array-representation of the concatenated text. In the online phase, we repeat the same process to get the embedding of the query and it is scored relatively to the documents embeddings using cosine similarity.

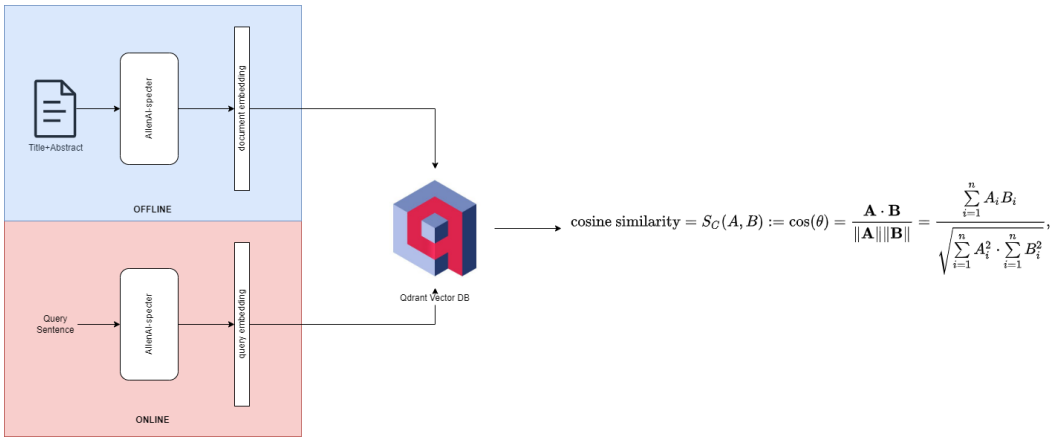

Figure 3: Usage in application

4.1 Why Sentence Transformers

The classical approach would be to tokenize text, build a vocabulary and an inverted index with each token to evaluate each document's TF-IDF score.
The problem with this approach is that it usually struggles with subwords and synonyms, so Word2Vec is a supervised-learned approach that uses language's statistics to associate to each token an embedding identifying it in a 512-dimensional array.
Word2Vec correctly deals with the various language-relative problems but it produces a single embedding for each word, so it averages over the various context the word is encountered in, moreover, the Word2Vec approach is only able to watch a fixed window of words, losing information about the entire document's text.
On the other hand Sentence Transformer exploits the attention system explained in the paper: "Attention is all you need" capturing a more detailed context that is aware of the entire text rather than a fixed amount of words.

4.1.1 Transformer Block

To explain the transformers in a few words: the input embedding of a token(word) is calculated and is added to its positional encoding so that the model is aware of the positional information about each word resulting in a positional-aware word embedding that is then used to compute three vectors:

- **Query**: an array representing the word as it was the query of an information retrieval system.
- **Key**: an array containing information about the content to which the token should pay attention to.
- **Value**: an array containing the information value of the token.

Using this information the transformer is able to calculate the relevance between tokens and compute the context out of an input text so that the output can be computed accordingly.

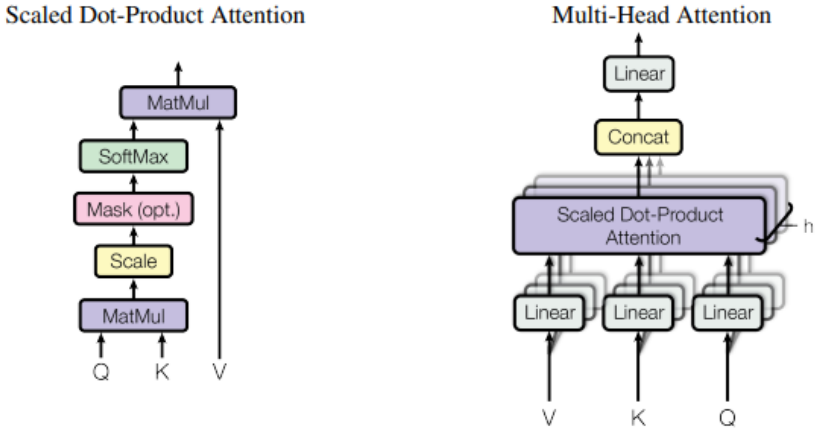

Figure 4: Attention-Head Layers

Each Attention Head will learn something about the language so that the final concatenation will produce consistent and generalized contexts out of the tokens represented by the Q-K-V matrixes.
The Q-K-V matrixes are able to recognize which parts of the inputs are relevant for the current token using an information-retrieval-based approach (dot product between query and key gives the importance of a query token with respect to a key representing another token in the input text).

With this in mind we can further analyze the Transformer architecture which is based on two main concepts:

- Self-Attention: An attention layer that is used on the input itself (the input learns to recognize what's relevant in itself to produce the required output).
- Positional encoding: It is used to generate position-aware embeddings so that the input is aware of each token's position in the input.

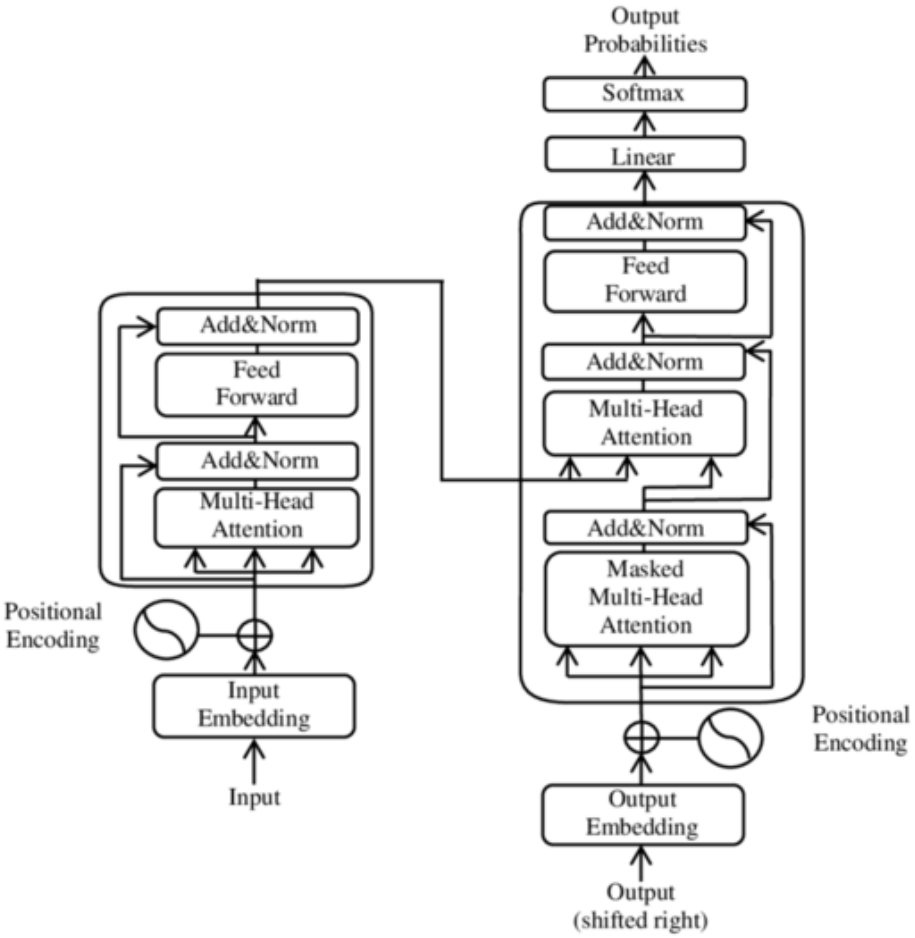

Figure 5: Transformer Block

In the above figure, we see the architecture of a Transformer block: the input is a sentence encoded as an array of embeddings. These embeddings are then added to positional embeddings which contain positional information of the content, encoded using sine and cosine functions. The resulting position-aware embedding is then split into the three matrixes (one array for each word token) Q-K-V and passed to the previously shown Multi-Head Attention Layer.

4.1.2 Sentence-Transformers

Sentence transformers are composed of a Tokenizer and a Transformer-based model followed by an average pooling layer that extracts the final embedding used for the representation of the document and the query. Those embeddings are compared using cosine similarity metrics to obtain a similarity score used for the final retrieval ranking. The advantages are that the tokenizer works together with the model so that only the tokens necessary to it are used for the computation of the embeddings so the text is automatically brought to the optimal state by the model's tokenizer. Moreover, the contextual information is extracted by the transformer which means that it will use the attention system with overcoming the window limitations as in the Word2Vec or the RNN approaches, catching synonyms and different meanings also for unseen words and misspellings thanks to the similarity with other words on which the transformer has been trained.

5 Model evaluation, summarization strategy and output analysis

To accomplish our summarization task, we chose as mentioned before an abstractive summarization approach, in particular with the choice of BART.
BART is a transformer encoder-encoder (seq2seq) model with a bidirectional (BERT-like) encoder and an autoregressive (GPT-like) decoder. It is pre-trained by corrupting text with an arbitrary noising function, and learning a model to reconstruct the original text.
The model we chose has been fine tuned on CNN-Daily Mail and XSum, a dataset that reports news found on BBC, providing the original document and a very short summary.
The version we chose is of the smallest available (230M parameters) in order to guarantee a faster inference and execution on common hardware, and is able to summarize 512-token texts into a user-defined size by using the most relevant words of the text or by generating new ones.

Supervised machine learning was applied to construct diagnostic models from a large, pooled university network database to predict COVID-19 upon suspicion or incidentally and thus, assist in clinicians' diagnosis before inpatient admission. A model that included clinical examination- and laboratory test features identified COVID-19 with satisfactory accuracy. Addition of chest CT features improved the model performance significantly.
Although RT-PCR tests provide an almost one hundred percent specificity, sensitivity, particularly in case of upper respiratory specimen, is not sufficient to rule out the disease [2]. Moreover, molecular laboratory-based RT-PCR test results are unlikely to be available before 24 h. Turnaround time of rapid point-of-care antigen tests is much shorter. Results should be available within 2 h of sample collection. However, sensitivity decreases with absence of symptoms, during the second week after symptom onset, and with no suspected epidemiological exposure. In addition, sensitivity varies with brands and probably varies with mutations that affect the virus nucleoprotein. Overall, sensitivity ranges from 34 % to 91 % in symptomatic and from 29 % and 78 % in asymptomatic individuals. Depending on prevalence, one in two to five true positives will be missed with rapid antigen tests [11]. Moreover, a recent Cochrane review revealed that absence or presence of individual signs or symptoms have only poor diagnostic accuracy to rule out COVID-19 [12].
Therefore, the syndromic presentation of COVID-19 as combination of signs and symptoms is better captured in a model based on a large dataset with a wide range of clinical, laboratory, and radiologic features as constructed in this study. Artificial intelligence can handle and analyze large datasets and shortens the procedure of model construction considerably. Machine learning algorithms develop real time prediction models that adapt to growing databases [[3], [5], [6], [7]]. Thus, diagnostic models that are based on machine learning may serve as important prerequisite to achieve readiness for surges of COVID 19 and other emerging or re-emerging pathogens.
The diagnostic model constructed in this study, is intended to rule out COVID-19 in a high-risk SARS-CoV-2 setting of hospitals. To protect vulnerable individuals at risk of severe disease, true positives must not be missed. Moreover, as care facilities can become amplifiers of infectious disease outbreaks, efficacious infection prevention and control is paramount (https://www.who.int/publications/i/item/WHO-2019-nCoV-Policy_Brief-IPC-2022.1). This gives reason to establish a highly sensitive test to be applied before inpatient admission [13]. On the other hand, false positives are not that critical because they can be identified with subsequent laboratory-based RT-PCR tests. Nevertheless, to avoid infection of false positives within the quarantine, patients who initially were diagnosed as positive by the model should be isolated separately until diagnosis is confirmed, so that false positives can be released from quarantine.
The most accurate model created in this study included features of all three categories (clinical, laboratory, and chest CT) and achieved a considerably higher sensitivity as established as minimum performance requirement (≥0.80) by the World Health Organization (WHO) for rapid diagnostic tests. Specificity of the model was sufficient for the desired purpose of "rule out", however, fell below the WHO requirement for rapid diagnostic tests (≥0.97)
(https://www.who.int/publications/i/item/WHO2019-nCoVAntigen_Detection2021.1).

Figure 6: Execution example of BART: highlighted in green, the concepts being summarized

For the sake of brevity, in this section we provide only one method and one set of tuning parameters, which are also the best found so far: they are however the result of an evaluation and considerations of the output.

5.1 Summarization strategy

Being our scientific papers long on average 5k to 10k tokens, they could not fit entirely into the input model, so a summarization strategy had to be chosen.
To address this problem we adopted a naive but yet efficient approach: each scientific paper was splitted into pages of 512 tokens, and then each page was summarized by BART, and then joined together into one single summary.

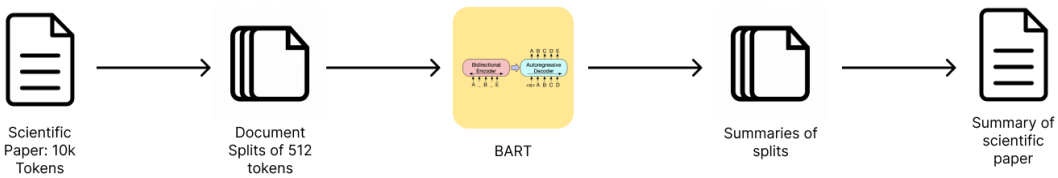

Figure 7: Summarization strategy

With this procedure we are able to avoid the input limit of 512 tokens for the summarization task.
The output size of the summary depends on a few parameters:

- Total number of tokens of the summary
- Number of tokens of the summary split
- Number of pages of the scientific paper obtained by splitting

In formulas:

$$T_s = \alpha \cdot T_{\text{tot}}$$

$$\alpha = T_{\text{max}} \div T_{\text{p}}$$

α: summary to reference ratio, which expresses the rate of text compression
T_{tot}: total number of tokens of the scientific paper
T_{max}: max token length of summary split
T_{p}: token length of paper split

5.1.1 Advantages and disadvantages of this approach

This kind of approach brings some advantages and disadvantages, let's start from the **advantages**:

- easy to implement: as this strategy regards splitting of a text, it becomes very easy to implement.
- easy to tune: there is only one parameter that we must take into consideration before performing the summarization task, the output size.
- bounded output size: as we know in advance how many tokens are present in each scientific paper, computing a higher bound of the number of tokens of the summary becomes very easy as explained before

Each page is obtained by truncation, so this brings some potential **disadvantages**:

- Context loss because of truncation
- poor control over the algorithm: short (but sometimes important) paragraphs can be ignored by summarizer

5.1.2 Summarizer (automated) evaluation

As discussed in the previous paragraphs, there are many methods available for the evaluation of a summary, however, most of them require an evaluation or a reference summary written by a human.
This kind of approach is very time consuming and requires an expert for each scientific, so this approach is not feasible.
For this reason we wanted at least to compare the quality of summaries generated by our model (BART) to a much more performant model which would then be used as a reference (GPT).
In this project we propose an automated method that, with the help of GPT APIs, tries to solve the discussed issues.

Here's the steps, which are represented in the graph below:

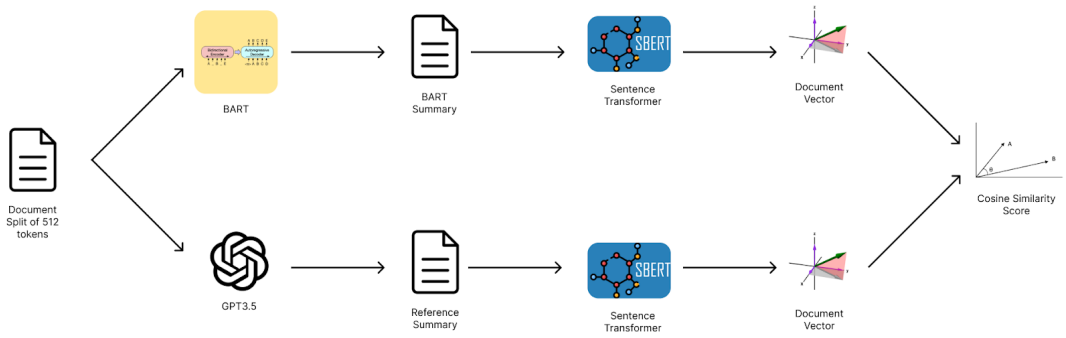

Figure 8: Summary evaluation process

1. retrieve the original full body text of the article
2. for each page (512 tokens) we perform summarization with BART and we ask GPT APIs to perform another summarization to obtain a reference summary
3. compare the reference summary with the summary obtained by BART and compute the cosine similarity metric (or potentially we could use any other metric such as ROUGE etc. . .) by vectorizing with a pre-trained sentence transformer fine tuned specifically on scientific papers
4. compute the average cosine similarity.

Using this approach and selecting 100 target papers we obtained an average cosine similarity between the reference summary and the obtained summary of 0.84, which can be considered sufficient for our purpose.
Using the same flow we measured a ROUGE-1 f-score and precision equal to 0.30 and 0.42 while ROUGE-L f-score and precision equal to 0.28 and 0.39.

5.1.3 Output analysis

In this paragraph we try to understand what the algorithm is actually summarizing, providing some output examples and understanding if the summaries can be considered sufficiently coherent for our purposes.
In yellow there's highlighted the concepts summarized by the algorithm, then reported below.

Original paper

1. Introduction

With the emergence of the coronavirus disease 2019 (COVID 19), pandemic health care facilities face the challenge to timely identify patients who are infected with the severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2). Rapid rule out of COVID-19 before inpatient admission is still crucial to prevent spread within high-risk transmission settings such as hospitals. Shortly after the pandemic onset, the reverse transcription-polymerase chain reaction (RT-PCR) test was available to identify SARS-CoV-2 from respiratory specimen. However, initially, diagnostic turnaround time amounted several days. Although, meanwhile, response time could be reduced to less than two hours with rapid diagnostic tests and to about

24 h with laboratory-based tests, sensitivity still depends on the viral load and thus, does not reliably rule out SARS-CoV-2 [[1], [2]].

Therefore, the COVID 19 pandemic provoked joint inter institutional research efforts to develop a diagnostic models as part of an infection prevention strategy that can rapidly and reliably rule out COVID-19 using data from routine clinical examination, laboratory tests, and chest CT. Models should accelerate and secure identification of patients with high probability of COVID-19, independent of RT-PCR test results, and thus, support decision making of physicians at the emergency department and other points of triage in favor of or against isolation of patients. Classification by such models could be both incidental and upon suspicion. An artificial intelligence approach using supervised machine learning for large datasets may become an efficient instrument to improve prospective pandemic preparedness [[3], [4], [5], [6], [7]].

The purpose of this study was to develop a predictive model using supervised machine learning, based on a university network database to identify COVID-19 in patients before inpatient admission. It should be assessed whether addition of chest CT features to clinical examination- and laboratory test features improves the performance of the diagnostic model. This approach could serve as a template to prepare for future health pandemics.

2.1. Study design

This study aimed to develop a machine learning-driven predictive model based on a large university network database to rule out COVID-19 among patients at emergency departments before admission to hospitals' regular wards to prevent in-hospital spread of SARS-CoV-2. The retrospective study was designed to show whether inclusion of chest-CT features to findings from clinical examination and laboratory tests improves the diagnostic performance of the model.

2.2. Study population and sites

A pooled database was retrospectively constructed from patient data, acquired at 12 German university hospitals between January 2017 and October 2020 (Supplemental Fig. 1). Participating sites included 4437 consecutive patients of at least 18 years of age (60.5 +- 23.2 years) who underwent chest-CT for any reason. A total of 692 (15.6 %) of the participants were SARS-CoV-2 positive according to the RT-PCR test (median proportion of COVID-19 positives in the centers: 13.2 % [IQR: 30.2 - 6.3 %]). Participants who were included before March 2020 were considered COVID-19 negative without RT-PCR test. Data were anonymized for analysis.

All 12 participating sites were part of the radiological cooperative network of the COVID-19 pandemic (RACOON) consortium. The RACOON consortium had been founded by 36 German university hospitals to establish an infrastructure to collect, transfer, and pool radiological data on COVID-19 for strengthening preparedness and responsiveness for pandemics. Data were acquired according to sites' routine standard of care from clinical examination, laboratory tests, and chest CT evaluation and collected using standardized RACOON electronic data capture templates. Investigators reported radiological findings using the mint Lesion™ software (Mint Medical, Heidelberg, Germany).

2.3. Model establishment

We constructed three prediction models to calculate individual participants' likelihood of being diseased

with COVID-19 and to classify participants as COVID-19 positive or negative using a supervised machine learning algorithm. The first model included only features from clinical examination and laboratory tests (model CL), the second model included only chest CT findings (model R), and the third model included variables from clinical evaluation, laboratory tests, and chest CT evaluation (model RCL). RT-PCR test was considered as reference standard.

For model construction, experienced specialists for internal medicine and radiologists of the participating sites manually selected 126 relevant candidate variables from the RACOON template as potential predictors for COVID-19 (19, 19, and 88 input features from clinical examination, laboratory tests, and chest CT, respectively [Supplemental Table 1]). To prevent overfitting, we conducted variance thresholding using a cut-off threshold of zero (Scikit-learn machine learning library, version 1.1.2,

https://scikit-learn.org/stable/) [8]. Subsequently, we used recursive feature elimination [9] with cross validation to select variables to be included into the following multivariable logistic regression analysis. During this process, the model was trained repeatedly while iteratively reducing the number of included features by removing the least essential features during each iteration. For evaluation of the model performance within an iteration, a stratified k-fold cross-validator with 5 folds was used, i.e., the model was split into 5 equally sized subsets and trained on 4 subsets while testing on the remaining subset. This process was repeated 5 times with a different subset being used as test set. The Scikit-learn machine learning library [8] was then used to run iterative logistic regression a hundred times using randomly generated training- and test subsets. With each run, 70 % of the data were randomly assigned as training dataset and 30 % as test dataset. The machine learning algorithm analyzed the training datasets to learn which variables are predictive of COVID-19 using the truncated conjugate gradient newton method to solve the optimization problem [10]. Weights were adjusted inversely proportional to class frequencies (balanced class weights).

Thirteen clinical examination, 10 laboratory test, and 7 chest CT covariables were identified as relevant features of the model RCL. The model CL identified 13 clinical examination and 10 laboratory test features, and the model R identified 8 chest CT features as relevant for classification Fig. 1 shows the association of every feature with COVID-19 determined with each of the models. The algorithm classified the event of COVID-19 as occurring if the probability according to logistic regression was $>= 0.5$. Classification was run with both the training datasets and the test datasets. The test datasets were analyzed to assess how accurately the algorithm predicted COVID-19 in the remaining 30 % of participants.

2.4. Statistical analysis

Diagnostic performance of the three models (R, CL, and RCL) was characterized by sensitivity, specificity, accuracy, negative predictive value (NPV), and positive predictive value (PVV). Receiver operating characteristic (ROC) analysis was performed and areas under ROC curves (AUC) were compared. Performance of the models was compared with z-test for paired samples. A difference of $p < 0.05$ was considered significant. Association of selected variables with COVID-19 was measured in odds ratios (OR) with 95% confidence intervals. Statistical analysis was performed with Python software (Phyton Software Foundation, Beaverton, USA, version 3.10.7).

3. Results

Overall accuracy in classification was 0.77, 0.84, and 0.89, respectively with model R, CL, and RCL.

Sensitivity was 0.87, 0.82, and 0.89, and specificity 0.75, 0.84, and 0.89, respectively with model R, CL, and RCL (results from the test datasets). The performance of model RCL that added chest CT features to the analysis, was superior regarding accuracy, sensitivity, specificity, NPV, and PPV compared to model CL that included only clinical examination- and laboratory test features ($p < 0.001$ for each of the outcomes referred) (Table 1). Chest CT features of ground glass opacity (RCL model: OR 2.69 [95 %CI: 2.65–2.74]) bronchus wall thickening (RCL model: OR 3.08 [95 %CI: 3.04–23.12]), and bronchiectasis (RCL model: OR 3.20 [95 %CI: 3.16-3.24]) contributed significantly to the prediction of COVID 19, whereas lung parenchyma mass > 30 mm and nodule as well as arterial occlusions reduced the odds of COVID 19. Chest CT features introduced the highest and the lowest odds ratios, i.e., the strongest associations with COVID-19, to the RCL model (Fig. 1).

The AUC of model RCL was significantly larger compared to model CL (difference in the test datasets: 0.030 [95%CI: 0.029–0.030], $p < 0.0001$) and to model R (difference in the test datasets: 0.073 [95%CI: 0.070–0.075], $p < 0.0001$) (Fig. 2). This means that the discriminative performance of model RCL to predict COVID-19 was superior to the models CL and R.

Confusion matrices show that proportion of false negatives (participants with COVID-19 who were wrongly classified as negative by the model) is significantly smaller with model RCL compared to the models CL and R (test datasets: RCL 1.7% vs CL 2.6% [$p < 0.0001$]; vs R 2.1% [$p < 0.0001$]). The same applied to the share of false positives (test datasets: RCL 8.8% vs CL 13.0% [$p < 0.0001$]; vs R 20.9% [$p < 0.0001$]) (Fig. 3).

4. Discussion

Supervised machine learning was applied to construct diagnostic models from a large, pooled university network database to predict COVID-19 upon suspicion or incidentally and thus, assist in clinicians' diagnosis before inpatient admission. A model that included clinical examination- and laboratory test features identified COVID-19 with satisfactory accuracy. Addition of chest CT features improved the model performance significantly.

Although RT-PCR tests provide an almost one hundred percent specificity, sensitivity, particularly in case of upper respiratory specimen, is not sufficient to rule out the disease [2]. Moreover, molecular laboratory-based RT-PCR test results are unlikely to be available before 24 h. Turnaround time of rapid point-of-care antigen tests is much shorter. Results should be available within 2 h of sample collection. However, sensitivity decreases with absence of symptoms, during the second week after symptom onset, and with no suspected epidemiological exposure. In addition, sensitivity varies with brands and probably varies with mutations that affect the virus nucleoprotein. Overall, sensitivity ranges from 34% to 91% in symptomatic and from 29% and 78% in asymptomatic individuals. Depending on prevalence, one in two to five true positives will be missed with rapid antigen tests [11]. Moreover, a recent Cochrane review revealed that absence or presence of individual signs or symptoms have only poor diagnostic accuracy to rule out COVID-19 [12]. Therefore, the syndromic presentation of COVID-19 as combination of signs and symptoms is better captured in a model based on a large dataset with a wide range of clinical, laboratory, and radiologic features as constructed in this study. Artificial intelligence can handle and analyze large datasets and shortens the procedure of model construction considerably. Machine learning algorithms develop real time prediction models that adapt to growing databases [[3], [5], [6], [7]]. Thus, diagnostic

models that are based on machine learning may serve as important prerequisite to achieve readiness for surges of COVID 19 and other emerging or re-emerging pathogens.

The diagnostic model constructed in this study, is intended to rule out COVID-19 in a high-risk SARS-CoV-2 setting of hospitals. To protect vulnerable individuals at risk of severe disease, true positives must not be missed. Moreover, as care facilities can become amplifiers of infectious disease outbreaks, efficacious infection prevention and control is paramount (https://www.who.int/publications/i/item/WHO-2019-nCoV-Policy-Brief-IPC-2022.1). This gives reason to establish a highly sensitive test to be applied before inpatient admission [13]. On the other hand, false positives are not that critical because they can be identified with subsequent laboratory-based RT-PCR tests. Nevertheless, to avoid infection of false positives within the quarantine, patients who initially were diagnosed as positive by the model should be isolated separately until diagnosis is confirmed, so that false positives can be released from quarantine. The most accurate model created in this study included features of all three categories (clinical, laboratory, and chest CT) and achieved a considerably higher sensitivity as established as minimum performance requirement ($>=0.80$) by the World Health Organization (WHO) for rapid diagnostic tests. Specificity of the model was sufficient for the desired purpose of "rule out", however, fell below the WHO requirement for rapid diagnostic tests ($>=0.97$) (https://www.who.int/publications/i/item/WHO2019-nCoVAntigen-Detection2021.1).

A recent Cochrane test-accuracy review that included 69 chest CT studies revealed a pooled sensitivity of 0.87 (range: 0.45–1.0) and a pooled specificity of 0.78 (range: 0.1–1.0). These findings led authors conclude that chest CT is appropriate to rule out COVID-19 but not to differentiate SARS-CoV-2 infection from other respiratory diseases [14]. Results concern only suspected cases. A French university study that reported on chest CT for rapid triage in multiple emergency departments also found favorable sensitivity (0.90) and specificity (0.88) [15]. However, due to radiation exposure even with a low-dose mode, chest CT should be justified by clinical indication. It should only serve as diagnostic approach in patients who require chest CT due to suspicion of COVID-19 or for whatever other clinical reasons. Of note, even in patients without characteristic symptoms, COVID-19 can manifest as pneumonia and signs can incidentally be recognized with chest CT [[13], [16], [17]]. Whether ultrasonography of the lungs may serve as alternative that might be applied more widely, even as screening tool without exposure to radiation, remains to be proven. The review mentioned above already found a similar sensitivity and a somewhat lower specificity for suspected cases (0.89 and 0.72, respectively, pooled from 15 ultrasonography studies) [14]. Thus, ultrasonography features may be considered in future predictive models. Another diagnostic approach could be radiomics methods. Based on artificial intelligence, imaging features can be converted into data for analysis and subsequently integrated into predictive models. Although initial findings are promising, to date, the approach of radiomics is not ready for clinical implementation [18].

Major strength of this study is the large database pooled from multiple nation-wide university centers that included consecutive patients who were admitted due to various diseases. COVID-19 positive participants could have been asymptomatic or symptomatic and there were also no restrictions regarding symptom onset. Therefore, we can claim a high degree of generalizability. Nevertheless, this study also has limitations. First, this study presents a static snapshot of the learning algorithm constructed. However, the algorithm may be continuously updated as additional data is acquired. Diagnostic performance is expected to improve with growing data volume over time. Moreover, in the course of time, the model can adapt to new virus variants. Second, we used RT-PCR test as reference standard for SARS-CoV-2 infection.

However, positive RT-PCR test results do not constitute infectiousness. Third, we only included patients who underwent chest CT. Although chest CT was conducted not necessarily due to suspicion of COVID 19 but rather due to various diseases, patient selection gives rise to sample selection bias. Finally, the test datasets originate from subdivisions of the pooled dataset. No external validation was conducted.

5. Conclusions

Chest CT features improve the performance of diagnostic models to predict COVID-19 before inpatient admission. An artificial intelligence approach of COVID-19 prediction can inform medical decisions right at the beginning of patients' diagnostic pathways in a timely manner. The approach might serve as an example of how to make use of large, pooled data bases to address future pandemics right from the beginning.

6. Declarations

Ethics approval and consent to participate

The study was approved by the Friedrich-Schiller-University ethics commission (Reg. No. 2021-2128). Data were anonymized for retrospective analysis and thus, ethics commission waived the requirement for informed consent.

Consent for publication

Not applicable.

Funding

This work was supported by the German Federal Ministry of Education and Research (BMBF) as part of the University Medicine Network (Project RACOON, 01KX2021). Authors' contributions

Summary obtained

P pandemic health care facilities face challenge to timely identify patients infected with severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2) Rapid rule out of COVID-19 before inpatient admission is still crucial to prevent spread within high-risk transmission settings such as hospitals.

Study population and sitesA pooled database was retrospectively constructed from patient data, acquired at 12 German university hospitals between January 2017 and October 2020. Participating sites included 4437 consecutive patients of at least 18 years of age (60.5 $\pm$ 23.2 years) who underwent chest-CT for any reason.

A stratified k-fold cross-validator with 5 folds was used for evaluation of the model performance within an iteration. The model was split into 5 equally sized subsets and trained on 4 subsets while testing on the remaining subset. This process was repeated 5 times with a different subset being used as test set.

Overall accuracy in classification was 0.77, 0.84, and 0.89, respectively with model R, CL, and RCL (results from the test datasets) The performance of model RCL that added chest CT features to the analysis, was superior to model CL that included only clinical examination- and laboratory test features ($p < 0.001$ for each of the outcomes

A model that included clinical examination- and laboratory test features identified COVID-19 with satisfactory accuracy. Addition of chest CT features improved the model performance significantly. Overall, sensitivity ranges from 34% to 91% in symptomatic and from 29% and 78% in asymptomatic individuals.

The most accurate model created in this study included features of all three categories (clinical, laboratory, and chest CT) This gives reason to establish a highly sensitive test to be applied before inpatient admission. On the other hand, false positives are not that critical because they can be identified with subsequent laboratory-based RT-PCR tests.

The study was approved by the Friedrich-Schiller-University ethics commission (Reg. No. 2021-2128) The approach of radiomics is not ready for clinical implementation. The approach might serve as an example of how to make use of large, pooled data bases to address future pandemics right from the beginning.

This work was supported by the German Federal Ministry of Education and Research. All authors reviewed and approved the final manuscript.

As we can observe, with this tuning (512 tokens page, 128 tokens summary) the algorithm seems to capture the concepts of the scientific paper without skipping too many important paragraphs, even if some of them are completely ignored.

Another thing that leaves space for future development is the fact that, creating a summary by concatenation of multiple summary splits, makes us obtain a summary that does not feel natural to read because of the presence of many sentences: there is a possibility that this might be improved with the usage of other generative networks or NLP techniques.

However, for the purpose of our application, the resulting summary provides the information needed to GPT APIs to answer the questions of the user.

6 GPT3.5

The final step in the application is to build a summary from a selected collection of papers, specifically built to answer the user's query. To make it possible, the GPT3.5-turbo model, with a context of 16k tokens, has been used due to its good trade-off between costs and performances.

6.1 Model

GPT is a Generative Pre-trained Transformer model, which means that it is able to generate textual content using information about the context given in the input prompt:

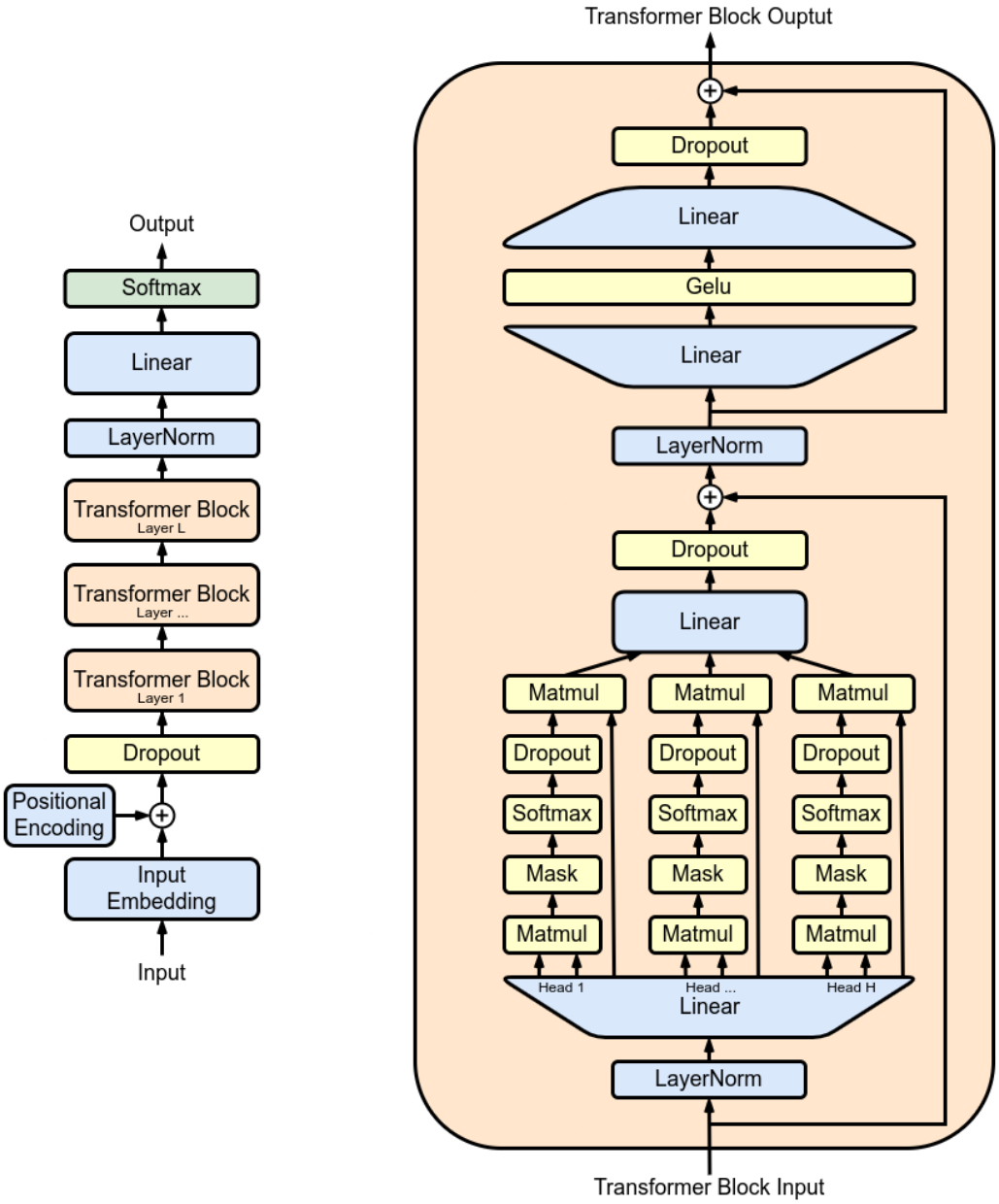

Figure 9: GPT Architecture

In each transformer block the input's context is furtherly analyzed and understood by the model, so that at the end GPT is able to produce content that will be fully aware of any hidden meaning and contextual information about the input text.

Below there is an example of how two different Attention-Heads learned different tasks for recognizing the context of each token:

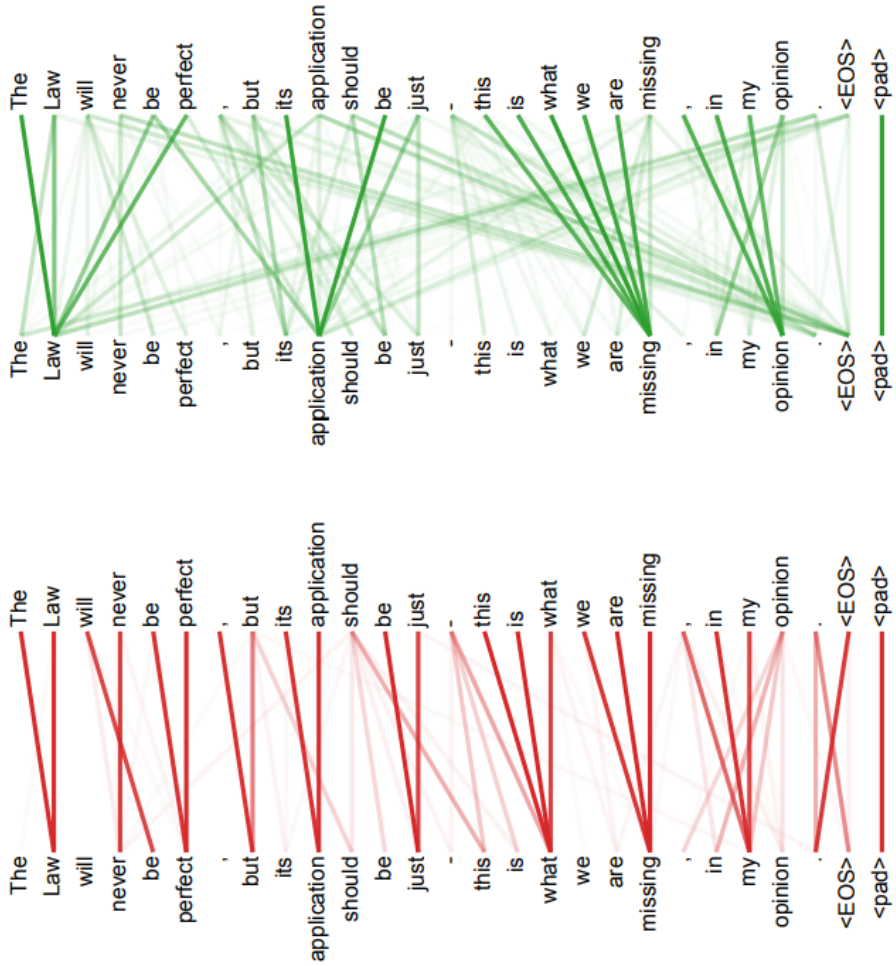

Figure 10: Two Attention Head giving different weights to tokens in the sentence

As the model advances in the output generation it will use a concatenation of the input text as well as the newly generated textual content enhanced with special tokens like the <EOS> one which is the "End Of Sentence" token to separate the different sentences and contents. In GPT3.5-turbo-16k this can be repeated up to 16 thousand tokens.

6.2 Usage in the Application

To obtain a summary from a set of selected papers, an API in Python has been developed, taking as input the query and the summaries of the selected papers, and through a process of prompt-engineering we developed a generic input prompt for GPT3.5-turbo-16k in order to induce it to produce a JSON document with a title and a content.

For the content, the prompt asks to merge the relevant information contained in the papers' summaries and reference them using the format [index]

```
prompt = f'You have been asked: "{query}" and you have information about the following papers summaries:\n'
for idx, doc in enumerate(results):
  prompt += f'index: {idx}; title: {doc["title"]}; summary: {doc["summary"]}\n'
  continue
prompt +=  f'.\nReferencing the provided articles with [index], \
  Generate a 4000 word article as a single json document with the following keys:\n \
  Title, Content.'
```

Figure 11: Input prompt to GPT

then a final "Reference" field is built in the final answer containing information about the index-to-link mapping.

```
section_text = f'<h1 style="text-align:center;">{answer["title"] if "title" in answer else "M
empty_stack = True
for section in answer:
  empty_stack = False
  if section == "title":
    continue
  section_text += f'<h2>{section.upper()}</h2>'
  section_text += f'<p>{answer[section]}'

section_text += "<h2>REFERENCES</h2>"
for idx, doc in enumerate(results):
  section_text += f'<p>[{idx}]\t<a href={doc["link"]} target="_blank">{doc["title"]}</a></p>'
```

Figure 12: Answer Building

6.3 Examples

how are iot wereable devices used for respiratory diseases?

The Use of IoT Wearable Devices for Respiratory Diseases

The Covid-19 pandemic in 2019 had a significant impact on global health, particularly in densely populated areas. As the virus primarily attacks the lungs, continuous monitoring of respiration is crucial for determining an individual's health status. In recent years, IoT wearable devices have emerged as a promising solution for monitoring respiratory diseases. This article explores the findings of several research papers on the use of IoT wearable devices for respiratory diseases.

One of the papers [0] discusses the development of a wearable graphene-based smart face mask (SFM) for real-time human respiration monitoring. The mask incorporates self-powered pressure sensors based on piezoelectric and electrostatic induction principles. These sensors provide continuous and accurate monitoring of respiration without the limitations of external power supply requirements. The SFM is equipped with strip-lines made of graphene, which act as strain sensor elements. These strip-lines demonstrate excellent response time and durability, making the SFM suitable for long-term daily use. The mask is also breathable, biocompatible, cost-effective, and user-friendly.

Another research paper [1] focuses on a face mask integrated with a flexible and wearable manganite oxide respiration sensor. The mask incorporates a sensing unit, a signal reading circuit, and a control chip. The flexible manganite oxide sensor demonstrates excellent sensing ability for human breath. The system wirelessly transmits the collected data to a smartphone or computer in real-time. Long-term tests have shown the stability and repeatability of the sensor in continuous breath monitoring. The wearable mask provides valuable information about human breath conditions, allowing for early detection of potential respiratory diseases.

Digital twin technology is also explored in a research paper [2] as a tool for revolutionizing healthcare systems. Digital twins are exact simulations of real-world objects or people and provide accurate and timely data for improved decision-making. In the healthcare sector, digital twins can enhance clinical operations and hospital administration. They enable quick prototyping of new technologies and provide precise forecasts and recommendations for individual patients. Digital twins offer real-time data analysis and can assist in identifying patterns and symptoms of respiratory diseases.

Overall, the use of IoT wearable devices for respiratory diseases shows great potential in improving the monitoring and management of respiratory conditions. The development of smart face masks and wearable sensors allows for continuous and real-time monitoring of respiration, providing valuable insights into an individual's health status. Furthermore, digital twin technology offers a new approach to healthcare, enabling personalized care and improved decision-making based on real-time data. As technology continues to advance, IoT wearable devices and digital twins will play an increasingly important role in respiratory disease management and prevention.

how is artificial intelligence used for respiratory diseases detection?

How Artificial Intelligence is Used for Respiratory Diseases Detection

Artificial intelligence (AI) has emerged as a powerful tool in the field of healthcare, including the detection and diagnosis of respiratory diseases. Several research papers have explored the application of AI in the detection of respiratory conditions such as acute pulmonary embolism (PE), sleep apnea, acute respiratory distress syndrome (ARDS) in patients with acute pancreatitis, and pulmonary tuberculosis in individuals with and without diabetes.

One study [0] focused on the development of a machine learning model using 12-lead electrocardiogram (ECG) for the diagnosis of acute pulmonary embolism. Pulmonary embolism is a life-threatening condition that often presents with non-specific signs and symptoms. The researchers used a single-center study at a tertiary university hospital, including a total of 1414 patients. They excluded 300 patients with SARS-COV2 infection diagnosis due to the known impact of the infection on D-dimer levels. The model was developed using PyTorch and trained using a NVIDIA 32GB V100S. The AI model demonstrated a specificity of 100% for the diagnosis of acute PE, which was significantly higher compared to other models.

Another research paper [1] focused on the classification of obstructive sleep apnea (OSA) severity using unsegmented peripheral oxygen saturation (SpO2) signals. Sleep apnea is a common sleep-related breathing disorder that can have serious health consequences. The authors developed a deep neural network-based model using state-of-the-art modules such as the cross stage partial network (CSPNet), SENet, and ResNet. The model was trained and tested using SpO2 signals recorded overnight. The model outperformed other counterparts in terms of classification accuracy, providing a reliable tool for assessing the severity of OSA.

In the context of acute pancreatitis (AP), another study [2] aimed to predict the occurrence of acute respiratory distress syndrome (ARDS) using machine learning models. AP is an inflammatory disorder that can lead to severe complications. The researchers used four machine learning algorithms to predict the risk for ARDS in patients with AP. They found that the Bayesian classifier (BC) model achieved the best predictive effect with the highest area under the curve (AUC) of 0.891. The model incorporated features such as white blood cell count, neutrophil-to-lymphocyte ratio, and calcium concentration, among others. Early prediction of ARDS can aid in the management and treatment of AP, improving patient outcomes.

One study [3] explored the use of AI for the detection of culture-confirmed pulmonary tuberculosis in individuals with and without diabetes. Diabetes mellitus is a significant risk factor for tuberculosis, and early detection is crucial for effective management. The researchers analyzed chest X-ray (CXR) images using a deep learning-based computer-aided detection (CAD) software. The study included 272 participants, and 23% of them had diabetes. The results showed that diabetes was associated with an increase in tuberculosis abnormality scores, particularly for smear-negative disease. The CAD software provided valuable insights for the detection of pulmonary tuberculosis in individuals with diabetes.

These studies highlight the potential of artificial intelligence in the detection and diagnosis of respiratory diseases. AI models can analyze various data sources such as ECG signals, SpO2 signals, clinical characteristics, and imaging data to provide accurate and timely assessments. The use of AI can improve the management of respiratory conditions, leading to better patient outcomes and more targeted treatment strategies. However, further research and validation are needed to ensure the reliability and generalizability of these AI models in diverse clinical settings.

7 Final results

7.1 System Interaction

In this section, we will describe the entire interaction with the retrieval system using as a query sample the sentence: *"how can artificial intelligence be used for agriculture improvement"*

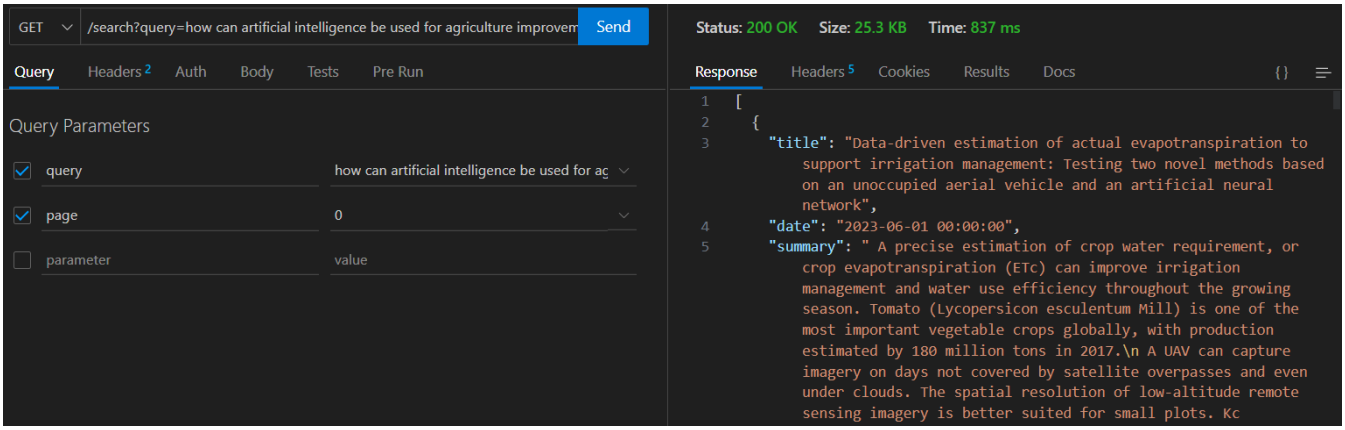

Figure 13: Thunder Client Performance Evaluation; The answer is provided as an array of JSON-formatted documents with the following fields: "title"; "date"; "summary"; "link"; "authors"

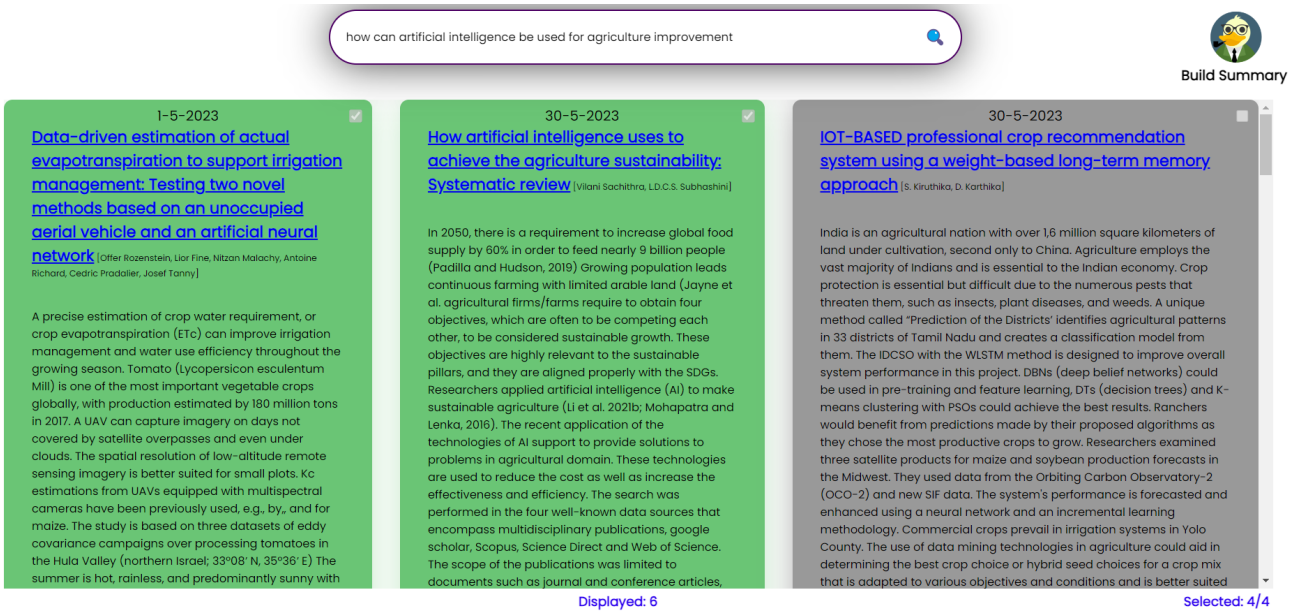

Figure 14: First and second selections for the final summary

In the above image, we see how the results are displayed, here we can see the search bar, to allow the user to change the query as needed and a "Build Summary" button for building the merged summary of the various

selected documents via GPT. The results are provided as cards containing the paper's title linked directly to ScienceDirect from where the scraping has been performed, the date on which the paper has been published, the authors that wrote the article, and the AI-generated summary, to let the user have a first glimpse over the content of each document.

From this section we selected the following 4 papers:

Title: Data-driven estimation of actual evapotranspiration to support irrigation management: Testing two novel methods based on an unoccupied aerial vehicle and an artificial neural network

A precise estimation of crop water requirement, or crop evapotranspiration (ETc) can improve irrigation management and water use efficiency throughout the growing season. Tomato (Lycopersicon esculentum Mill) is one of the most important vegetable crops globally, with production estimated by 180 million tons in 2017.

A UAV can capture imagery on days not covered by satellite overpasses and even under clouds. The spatial resolution of low-altitude remote sensing imagery is better suited for small plots. Kc estimations from UAVs equipped with multispectral cameras have been previously used, e.g., by,, and for maize.

The study is based on three datasets of eddy covariance campaigns over processing tomatoes in the Hula Valley (northern Israel; 33º08' N, 35º36' E) The summer is hot, rainless, and predominantly sunny with slight variation from day to day, and winter is cool and rainy.

T19b and T20 campaigns were conducted at Kibutz Gadot, in two adjacent commercial fields in 2019 and 2020.

A combined temperature-humidity sensor measured air temperature and humidity (HMP45, Campbell Sci., Logan, UT, USA) and soil thermocouples. Each dataset consisted of half-hourly latent heat flux (LE; associated with ETc) and meteorological variables measured at the same tower.

Every week or two, around noontime, the leaf area index (LAI) was measured in two areas in the field, averaging 30 readings in each area, using Sunscan-SS1 (Delta-T Devices Ltd, Derbyshire, U.

Micasense imagery was processed into orthomosaics using Pix4Dmapper (Pix4D S.A., Prilly, Switzerland). Level 2A corrected Sentinel-2 Images were downloaded from the Copernicus Open Access Hub website. Satellite and UAV images were then resampled using bilinear interpolation to 10m resolution.

The network takes as an input two crop variables: the leaf area index (LAI) and days after planting (DAP) The LAI was acquired from ground measurements using the Sunscan-SS1 (Delta-T Devices Ltd, Derbyshire, U.

Despite using the weather stations 'Hava 1' and 'Gadot' for the analysis of the EC data, it was decided to use the 'Kavul' station as the source of the meteorological data for training the ANN for several reasons.

An irrigation trial was conducted in Gadash farm, close to the previous data-collection campaigns. Processing tomato cultivar H-4107 was transplanted with a plant density of 25,000 plant ha-1. After transplanting, the entire field was irrigated with 30mm of water in order to fill the soil profile.

A buffer masked every treatment replicate to include only pixels that were 2m away from an adjacent treatment.

Three metrics were used to evaluate the performance of the different irrigation methods: yield, water use efficiency, and brix. The yield is the most important metric; it is a measure of the fresh biomass of harvested red tomato fruit per unit area (e.g.

A possible reason for the slightly lower energy balance closure and R2 in Gadot 2020 is frequent disturbances to the net radiometer, IRGA and sonic anemometer caused by large presence of birds. There is a clear resemblance between LAI and Kc courses with similar LAI to Kc differences among the seasons.
Standard Kc also preceded Gadash 2019 in about a week, and values were generally higher than Gadash except at the end of the season, where the measured Kc exceeded the standard table value.
Anomalous yield results in the North Western part of the field (Fig. 11) were probably due to wind direction. During summer, the prevailing wind direction in that region is northwest. It is hypothesized that the wind carried away water vapor from the plants' canopy close to the upwind edge. This is expected because the rate and dosage of irrigation were similar throughout the experiment.
The 50% treatment was slightly higher than the rest, but this difference was not statistically significant. The reason for that is the inverse relationship between yield and quality depending on the irrigation regime. Brix levels were generally lower than average in Israel in recent years, around 4.8.8.
Remote sensing provides a better spatial representation of the field compared to point measurements or a single tabular value. Kc maps might offer a better alternative for farmers like the one in Gadot that avoided the use of a single tensiometer that does not represent well the entire field.
The claim holds merit, particularly in cases of extensive, low-frequency irrigation of deep-rooted crops grown in fine- texture soils. In both cases, estimates of ETc as the required irrigation dose would not be optimal. We conclude that ETc estimates must be supplemented by a more complex model that considers the water movement in the soil.
AV approaches almost perfectly agreed with the best practice, both in the total amount and rate of irrigation throughout the season. Future work should replicate the experiment to establish further the experimental approach for irrigation management.

Title: How artificial intelligence uses to achieve the agriculture sustainability: Systematic review

In 2050, there is a requirement to increase global food supply by 60% in order to feed nearly 9 billion people (Padilla and Hudson, 2019) Growing population leads continuous farming with limited arable land (Jayne et al.
agricultural firms/farms require to obtain four objectives, which are often to be competing each other, to be considered sustainable growth. These objectives are highly relevant to the sustainable pillars, and they are aligned properly with the SDGs.
Researchers applied artificial intelligence (AI) to make sustainable agriculture (Li et al. 2021b; Mohapatra and Lenka, 2016). The recent application of the technologies of AI support to provide solutions to problems in agricultural domain. These technologies are used to reduce the cost as well as increase the effectiveness and efficiency.
The search was performed in the four well-known data sources that encompass multidisciplinary publications, google scholar, Scopus, Science Direct and Web of Science. The scope of the publications was limited to documents such as journal and conference articles, published in English.
AI methods use in agriculture are one of the emerging areas of research in recent generation. Today AI is used to solve the problems particularly to reduce the use of the labor force, to enhance efficient utilization of resources and to facilitate the development of sustainable business. With the rapid technological

advancement, people are more intend to develop these applications.
Artificial neural networks (ANN) are one of the most important technique of AI. These models are developed using interconnected nodes which are performed functions as our human brain.
DL has applied to identify seeds and pest, monitor nitrogen content in soil and leaf, detect irrigation and plants' water stress level, assess erosion of water, detect use of herbicide, detect usage of herbicides and monitor greenhouse. However, DL models need comprehensive datasets as the input to serve at the training procedure.
The ANN and DP techniques are used to collect real-time data about multiple agricultural parameters, such as production quantity, waste, climate data, biomass, and land area. The fact that this AI application in prediction model in agriculture is so common can be justified by the complex and dynamic nature of the agricultural parameters.
Computer vision is used for crop ripeness estimation (Hespeler et al., 2021). In all cases, crops should be picked when they are ripe or mature without mechanical damage to the fruit. This action should take place as quick and as cost-effective as possible.
Symptoms of diseases developed by attacks of bacteria, fungi, and other pests need to be identified in an initial stage according to the changes in the physiological condition of plant parts (leaves, stems, and flowers) The robotic disease-detection systems were commonly designed in whole inclusive pattern to identify the results in infection and these results could be utilised to detect precise diseases and apply fertilizes appropriately.
The review emphasized that AI driven agriculture is focusing on methods to optimize land. Using AI in weed control enables to decrease unnecessary plants within fewer time frames and minimize fertilizers and herbicides utilization.
The review insists that AI applications, especially ML, in agriculture SC enable farmers and other relevant organizations to draw valuable insights on agriculture process. Data pays a crucial role in supply chains thus improvisation in storage, collection, visualization, privacy, security, accuracy and access of agriculture data can impact application of AI in agriculture supply chain.
Review paper focuses on how AI technology can improve the sustainability of agriculture industry. The aim of this review paper is to analyse and create an understanding of the different types of AI applications in agriculture industry and how those applications align to achieve the agriculture sustainability objectives.
Weeds destructively affect agricultural crop productions by contending with crop plants for resources, including soil moisture and nutrients. Such complex operation process might slow down data acquisition and integration, leading to an information lag. This implies the need for international harmonization and standardization in phenotyping data.
The most common applications of AI for agriculture are prediction model for total agricultural output value, followed by harvesting applications. In recent work the use of AI and image processing techniques has become more common to improve the sustainable agriculture. In future researchers can consider developing attention-based DL models.
S. Subhashini: Conception and design of study, Acquisition of data, Analysis and/or interpretation of data.

Title: A systematic analysis of machine learning and deep learning based approaches for identifying and diagnosing plant diseases

The abiotic variable is a biological factor such as bacteria, fungi, or algae, as opposed to factors such as rain, moisture, and temperature. Methods for automating the detection of plant leaf diseases have been developed using artificial intelligence, machine learning, and Deep learning. In agriculture, deep learning is the most widely used application.
The Support Vector Machine (SVM) uses a linear classifier as the basis for learning. Support vectors on both sides of the plane are aware of nearby hyperplane locations. The purpose of this study is to review the existing research status and opportunities for future research concerning Industry 4.0 technologies.
Industry 4.0 is one of the most significant aspects of any organization, industry, or nation. There are several applications that are possible, and its implementation will change the workplace in several ways. This paper will be arranged similarly to Section 2, which is a list of numerous previous studies that have been conducted in this field.
Machine-learning techniques are becoming increasingly popular for identifying disease in plants. An early diagnosis of crop health and disease can be achieved by applying appropriate management approaches, such as fungicide sprays, drugs for specific diseases, and pesticide-based vector control. Identifying healthy and diseased leaves is essential for controlling crop loss and boosting yield.
Accuracy, precision, recall, and F1 score are metrics used to compare the proposed method to five machine-learning classifiers. Accuracy is the proportion of samples in the overall dataset that were correctly categorized. PrecisionFalse-negative occurs when a test for common rust is negative but the plant still has the disease.
The termination error rate represents the maximum allowable mistake in classifying values in a neural network. The network's efficiency is optimal for a high termination rate. The higher the termination rate, the better the neural network's performance. The system's capacity to correctly categories samples into their appropriate classes can be used to assess its performance.
The MHGSO-optimized DenseNet-121 architecture is fed the test pictures, and the error rate is compared to the real class. Following each iteration, the model's behavior is calculated based on the loss value.
A deep learning algorithm trained with the plant village dataset can identify leaf diseases accurately. The accuracy of classification can be improved through data augmentation, large datasets with high variability. The authors hope that extending this research will result in a valuable contribution to sustainable agriculture.

Title: Evaluation of IoT based smart drip irrigation and ETc based system for sweet corn

In order to reduce water consumption in agriculture, real-time smart irrigation technologies are needed. In a few studies, evapotranspiration (ET), soil moisture sensors (ET) or plant-based smart irrigation technology have been used to schedule irrigation events by considering the weather, soil moisture conditions, and plant water status.
Sweet corn (Zea mays var.
The IoT device is enclosed in an IP-65 weatherproof box, which allows it to be securely installed on the study site. Six similar devices were installed in the 300 m2 field during initial testing to give spatial resolution for data. The collected data from the sensors is transferred to a cloud server (ThingSpeak) by

the device once the system is established.
Six IoT data acquisition systems were placed at different locations in the field. Moisture values were recorded every two hours between 8:00 am and 6:00 pm. The oven-dry method was also used to determine soil moisture content. The following mathematical relationship was developed based on soil moisture readings from sensors.
The experimental design was a Tukey test statistics with an analysis of variance (ANOVA) was used. The analyses were carried out in order to find any significant differences between the IoT and ETc based drip irrigation treatments. In general, sensors placed deeper in the soil profile result in lower irrigation efficiencies, which leads to more frequent irrigation.
In the IoT (soil moisture-based) and ETc-based treatments were compared at 10-day intervals during the growing season. The treatment ETc100 per cent received the most water, followed by SM43.5 per cent and SM34.8 per cent. The amount of water-saving was observed as 11% and 23.
The ETc100% treatment had a low grain and stover WP due to a low yield; this was true for all treatments. This might be due to the crop using water more efficiently, and smart moisture sensors triggered irrigation in real-time based on predefined thresholds during the growth season, resulting in optimal water use and better yield. The IoT-SM43.
Ph. D student, conceptualized, carried out the experimental work, and prepared the manuscript, data curation, software, written- original draft preparation.

From those the final result generated through GPT is shown below:

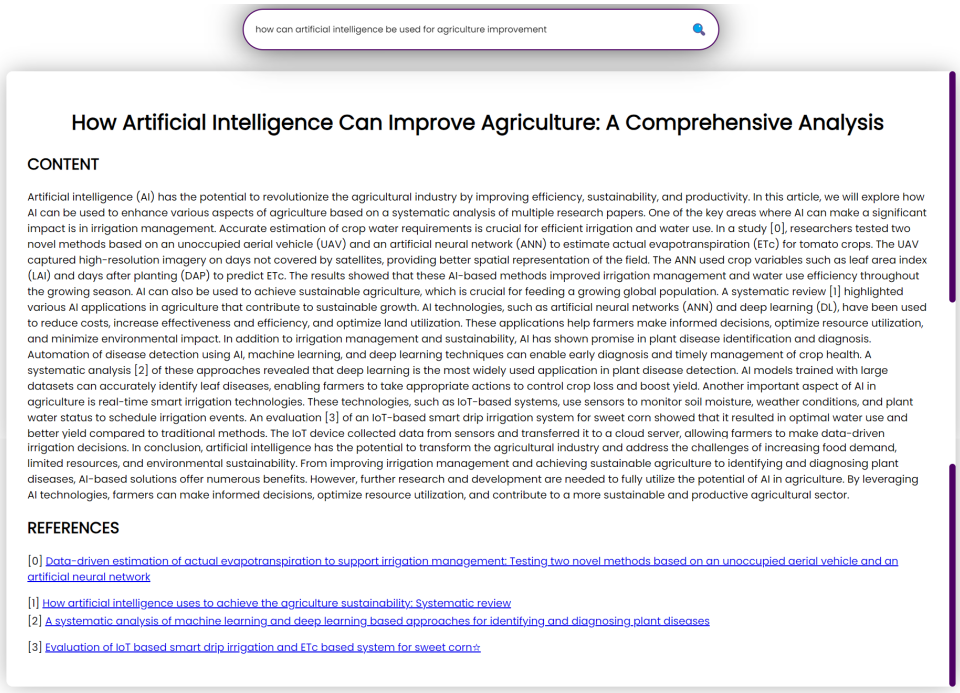

Figure 15: GPT-Generated Merged Summary

7.1.1 Retrieval Speed Performances

For evaluating the speed performances we used Thunder Client which allows to test an API and also gives information about the time to get the answer.
Below is a table with the query and its relative time in milliseconds:

QUERY	TIME [ms]
impact of IoT systems on environment	811
how can artificial intelligence improve driving systems on transport means	840
what can be used to build a deep learning model in python that measures the intensity of pollution in the environment knowing parameters like the air composition at a given time of the day, and how can it be used to extract analytics	832
security management state of the art	756
Impact of climate change on coastal ecosystems: A comprehensive review	795
Neuroplasticity mechanisms in cognitive rehabilitation after traumatic brain injury	734
Ethical considerations in artificial intelligence: Balancing innovation and societal implications	743
The microbiome-gut-brain axis: Current insights into its role in mental health disorders	853
Advancements in renewable energy storage technologies for a sustainable future	755
Exploring the link between social media use and adolescent mental health: Recent empirical studies	754
Precision medicine approaches for personalized cancer immunotherapy	741
Urban planning strategies to enhance resilience against natural disasters: Case studies from megacities	850
Effects of intermittent fasting on metabolic health and longevity: A systematic meta-analysis	809
Unlocking the potential of CRISPR-Cas9 gene editing: Challenges, breakthroughs, and ethical implications	749

As emerged from the below's results, thanks to the usage of fixed-size embeddings, one crucial advantage of this approach is that the retrieval time is completely independent of the query prompt length and number of stopwords. The embedding size, in fact, impacts the time of the answer as well as the relevance of the retrieved documents, for this reason, Allenai-specter has been chosen: it generates 768-sized embeddings which is a

good compromise between speed and relevance for our system, moreover it is specifically fine-tuned on research articles, so it well fits the scope of our application.
To further prove the independence of the query prompt from the retrieval time, we generated an overlong query using ChatGPT, the results are shown below:

QUERY

An in-depth analysis of the multifaceted impacts of anthropogenic climate change on coastal ecosystems, encompassing changes in sea level rise, ocean acidification, temperature variations, and extreme weather events, and their interplay with biodiversity loss, habitat degradation, and socio-economic vulnerabilities. A comprehensive review is sought, elucidating the intricate mechanisms underlying neuroplasticity in the context of cognitive rehabilitation following traumatic brain injury, exploring synaptic modifications, neurotransmitter dynamics, and network reorganization. Investigating the intricate ethical landscape surrounding artificial intelligence, this query aims to uncover scholarly discussions on striking a balance between innovation and the potential societal, economic, and ethical ramifications. Seeking a current synthesis of research into the microbiome-gut-brain axis, with a focus on its role in mental health disorders, considering bidirectional communication, microbial diversity, and emerging therapeutic interventions.
Furthermore, an exploration into the latest breakthroughs in renewable energy storage technologies is needed, with an emphasis on scalability, efficiency, and integration with existing grids, underscoring the critical role of such advancements in achieving a sustainable energy future. Investigating the complex relationship between adolescent mental health and social media use, the query seeks to identify recent empirical studies dissecting causality, moderators, and potential interventions, contributing to a nuanced understanding of this contemporary concern.
Delving into the realm of personalized cancer immunotherapy, a comprehensive review is sought to illuminate the latest strides in precision medicine approaches, encompassing tumor profiling, immune checkpoint inhibitors, neoantigen discovery, and clinical outcomes, with a view to enhancing treatment efficacy. In the context of escalating urbanization and increased vulnerability to natural disasters, this query aims to uncover urban planning strategies that foster resilience, featuring case studies from megacities worldwide, shedding light on infrastructure design, community engagement, and policy frameworks.
Investigating the effects of intermittent fasting on metabolic health and longevity, this query demands a systematic meta-analysis integrating diverse studies on fasting protocols, physiological adaptations, and underlying molecular pathways, critically evaluating the potential benefits and risks. Lastly, delving into the forefront of genetic engineering, this query seeks to unravel the multifaceted landscape of CRISPR-Cas9 gene editing, encompassing technical challenges, recent breakthroughs, ethical considerations, and regulatory perspectives, contributing to a comprehensive understanding of its transformative potential.

RESPONSE TIME: 985ms

7.2 Limitations of this project and suggested improvements

As we showed in previous paragraphs, we encountered several limitations not only in terms of time but also in resources and technology: these limitations leave space for future improvements and new suggestions, which in this subsection will be described in detail for each task/component of our application.

7.2.1 Limitations of free to use generative models

One of the biggest issues encountered during the development of the project was the limitation in input size of free to use models. The free versions had a maximum token limit for inputs that was not suitable for either the summarization process or for generating comprehensive summaries.
In the context of summarization, the input text often contained a significant amount of content that needed to be distilled into a concise summary. This meant that longer documents with rich information could not be effectively processed by the free models due to their token limits. As a result, the summarization process would either be cut short or the model would struggle to capture the essence of the content (this led to the choise of a summarization strategy).

7.2.2 Summarization task

In our project we obtained decent results in the summarization task, but still, in some papers the summarizer is not able to fully abstract some sentences and acts as an extractive summarizer, probably this can be improved by a fine tuning phase or by the usage of a different summarization strategy: being this phase very expensive in terms of resources we decided to skip it.
We could have used GPT directly for each summarization, to achieve a high quality result, but it would mean a huge expense in terms of credits.

7.2.3 Summarization strategies

In this project the simplest summarization strategy was used (splitting), and as mentioned in the related paragraph, it has many drawbacks: new strategies must then be proposed, for example recursively splitting by paragraph, which can keep more context and improve summarization quality.

7.2.4 GPT

The main problem with GPT is that each query consumes credits, so we had to be very careful to not over use this API.
Prompt must be engineered and improved or an NLP approach must be used (if available today?) in order to have a more fluent article, there are still too many sentences.
To achieve this a higher number of tokens input would be needed, which can make us end up with no tokens left for a complete query to GPT or with the need to perform multiple queries to GPT.

7.3 References

- EVALUATION MEASURES FOR TEXT SUMMARIZATION

- METHODS FOR COMPARING RANKINGS OF SEARCH ENGINE RESULTS
- TECHNIQUES: https://towardsdatascience.com/a-quick-introduction-to-text-summarization-in-machine-learning-3d27ccf18a9f
- BART
- ATTENTION IS ALL YOU NEED