Logo

Description automatically generated A picture containing text

Description automatically generated

Text Summarization Using Deep Learning

|  |  |
| --- | --- |
| Name | ID |
| Ali Amr Nour Mokhtar | 20190339 |
| Kerllose George Jack | 20190393 |
| Sohaila Mahmoud El Sayed | 20190250 |
| Mohammed El Mogy | 20190473 |
| Menna Allah Fathy Lashin | 20190560 |
| Youssef Hesham | 20190647 |

Under the Supervision of:

Dr. Motaz El Saban

Eng. Belal Tarek

Artificial Intelligence Department

Cairo University

2022-2023

Table of Contents

1. Abstract
2. Introduction
3. Scope
4. Problem Statement
5. Objectives
6. Related Work
7. Background
   1. The Transformer
   2. Autoencoding Transformers
      1. BERT
      2. ALBERT
      3. ELECTRA
   3. Autoregressive Transformers
      1. T5 and mT5
      2. BART and mBART50
      3. Comparison Between mT5 and mBART50
   4. Models Worth Mentioning
8. Current Work
   1. Arabic Specific Models
   2. Our Path
9. Datasets and Metrics
10. Results
11. Web interface
12. Other Possible Applications
13. Future Work
14. References

**Abstract**

In this document we discuss the topic of text summarization using deep learning in full details, we describe the history of the problem and its models, the current methods, and techniques, and we discuss the path we have taken to get the best model possible.

**Introduction**

The task of text summarization is the computational compression of an input reference into a summary that is representative of the input reference.

Possible subtasks include but are not limited to:

* Abstractive Text Summarization, which could be described as the summarization of text by teaching the model to process the input article and generate a summary that may contain new words/sentences/phrases that were not previously present in the input article.
* Extractive Text Summarization, which may be defined as summarization through teaching the model to separate the input reference article into sentences, to rank the sentences by their quality as representative of the input article, then choosing and concatenating the top sentences to form a coherent summary.
* Scientific Document Summarization, which may be defined as the summarization of scientific material.
* Multi-Document Summarization, which could be described as the process of representing a set of documents with a concise piece of text through capturing the relevant pieces of information while filtering out the irrelevant ones and could be done abstractly or extractivly.

This task has been a very strong challenge for models across the years, it is a difficult task for models to master, and mastering it is a telling sign of the strength of a model.

**Scope**

This thesis focuses on the types of models historically and contemporarily used, the difference between them architecture-wise, and state-of-the-art (SOTA) techniques.

It also covers the path that we have taken in search of the best model.

**Problem Statement**

Text summarization is the task of producing a document that contains the most information possible while having a fraction of the length of the original document.

**Objectives**

This thesis aims at describing, explaining, and presenting the historical and contemporary methods of performing the task of abstractive text summarization, with an emphasis on the current SOTA methods.

The reason behind choosing this task is its practical viability, the idea of a programme that summarizes text in a human-like manner while being faithful to the original content of the document is quite attractive to a large segment of the population, especially those who have jobs that require going through large amounts of documents on a frequent basis.

**Models and Their History**

Historically speaking, the task of text summarization has been performed in a various number of ways, the paper by Qaroush et. al. [[1](#ref1)] which include:

* Semantic-based summarization
* Statistical-based summarization
* Traditional Machine learning-based summarization
* Cluster-based summarization
* Graph-based summarization
* Discourse-based summarization
* An optimization-based summarization

**Semantic-based Summarization**

Semantic analysis deals with the intrinsic meaning of the words as well as the connections/relations between words, phrases, and sentences to represent and capture the intended concepts of the text.

Various semantic analysis techniques can be applied to summarize texts including lexical chains and natural language processing methods such as latent analysis which can be used in addition to statistical techniques.

**Statistical-based Summarization**

Statistical approaches have been widely used in text summarization, the concept of relevance score which depends on the extraction of a set of features is the decisive factor that reflects the importance of a sentence regardless of its meaning.

The extracted key-phrases can be based on some features like Term Frequency (TF), inverse document frequency (IDF), font types and their existence in the document title. The extracted key phrases are then assessed to their ability to reflect sentence importance.

Some researchers have used other features to score the sentence including indicator phrases, uppercase words, sentence length, similarity with the title, and sentence position in the document.

Meanwhile, other researchers have used a weighted linear combination of statistical features for sentence ranking, while obtaining the optimal weights using a genetic algorithm.

Using statistical features alone might not provide good results, because they do not take into consideration the meaning of the words and the relations between them as well as the relations between the sentences themselves. Furthermore, another expected problem is redundancy in the selected sentences.

**Traditional Machine learning-based Summarization**

Historically speaking, machine learning has been used in summarization using the traditional techniques such as SVMs, neural networks, and Gaussian mixture models.

**Cluster-based Summarization**

The Clustering process aims at grouping objects into classes drawing on similarities. While summarizing texts, the objects are the sentences, the classes are the clusters that the sentences belong to. In this approach, the formation of the summary is performed by selecting a sentence or more from each cluster based on the closeness to their cluster centroid.

**Graph-based Summarization**

In this approach, the document is illustrated in a graph like the model. In this model, the nodes of the graph represent the sentences, while the links/edges between the connected nodes represent the similarity relation between sentences. Therefore, a sentence is considered important if it is strongly connected to many other sentences.

**Discourse-based Summarization**

In this structure, instead of treating the text as a continuity of words and sentences, texts are represented or organized in a way where discourse-units are related to each other to ensure both discourse coherence and cohesion. Building successful discourse structures mainly depend on the availability of robust discourse parsers which rely on four factors including the type of discourse theory, the data structure used for representing structure (tree, or graph), the nature and the hierarchy of the relations (semantic, intentional, or lexically grounded) and finally the language.

**An optimization-based Summarization**

Text summarization is considered by many researchers as a Multi-objective optimization problem, where a set of objectives are considered to produce a high-quality summary including coverage, redundancy (diversity), coherence, and balance.

Coverage means that summary should contain all important aspects appearing in the documents, while diversity aims to reduce the similar sentences in the output summary.

On the other hand, coherence aims to generate a coherent text flow. Moreover, balance means that summary should have the same relative importance of different aspects of the original documents.

By far the machine learning based approach, especially the deep learning one has proven itself to be the most effective and most powerful approach to summarization, with the SOTA methods being the transformer architecture.

**Background - Transformer-Based Solutions**

The transformer architecture [[2](#ref2)] currently dominates the NLP, Computer vision [25] and, increasingly, the time series [[23](#ref23)], graph representation [[24](#ref24)] learning, and reinforcement learning [[22](#ref22)] domains, such that it is explicitly agreed upon that RNN architectures are mostly dead.

RNNs are:

* Computationally expensive, recurrent has a time complexity per layer of , which is disadvantageous in NLP, as the data usually has a small length and a large number of dimensions.
* Hardware unfriendly.
* Susceptible to gradient issues.
* Hard to train due to their recurrent nature.
* Time consuming to debug due to their complicated nature.

Meanwhile transformers are:

* Computationally inexpensive, attention has a time complexity per layer of which is advantageous in NLP, as the data usually has a small length and a large number of dimensions.
* Highly parallelizable.
* Vastly less prone to gradient issues.
* Much simpler architecture.

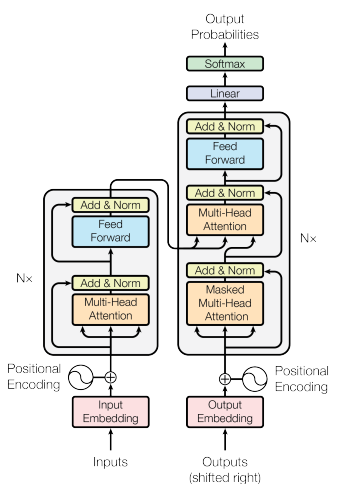
**The Transformer**

The Transformer is the earth-shattering neural architecture that is currently dominating the AI field, the transformer has many traits that make it so outstanding, of which is the global receptive field of the Multi-head Self Attention module, which is quite lucrative for extracting global information and dependencies.

Moreover, the transformer is a highly parallelizable architecture, it can be divided using both model parallelism and data parallelism, due to its GPU friendly nature, unlike its predecessor the RNN, which is purely sequential in nature.

The transformer is very likely to become more efficient as time passes by, due to the large amount of interest it has garnered since its release.

The main issue with the architecture is the self-attention module itself, it is unfortunately, computationally expensive at the large scale the SOTA models have reached, and researchers are racing to optimize it.

**Architecture**

The architecture itself consists of

The following elements:

* Positional Embeddings
* Multi-Head Attention (MHA)
* Feed Forward (MLP)
* Masked Multi-Head Attention (MMHA)

**Positional Embeddings**

Due to the parallel nature of the transformer, sequential information and positioning is lost, and thus the authors of the transformer paper introduced the idea of a positional embedding.

The idea behind it is to assign a position to each word as a function of feeding the embedding vectors to a sinusoidal wave function which establishes the sequence between the words without overwhelming the weights and causing gradient issues.

Where Pos is the position of the word, i is the current element in the vector, and d is the embedding size.

Other methods were tested, such as learned embeddings, but the Authors chose the sinusoidal version, they reasoned “because it may allow the model to extrapolate to sequence lengths longer than the ones encountered during training.”

**Multi-Head Attention (MHA)**

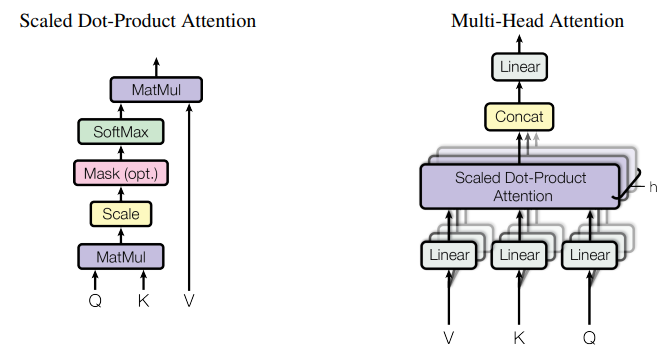
It is the heart of the transformer and the secret of its success, it is inspired by previous work on attention mechanisms, but here in this context it is *self*-attention, where each value in the tensor attends to each other value.

Attention mechanisms, generally speaking consist of dot products followed by a SoftMax, and such is the case, except that the Query, Key and Value are the same tensor,

The following is the Scaled Dot-Product Attention.

The scaling factor is to counteract gradient issues.

Fully connected layers are used before every layer and after the concatenation of the outputs of the heads.



**Feed Forward (MLP)**

A normal combination of three fully connected layers with a large hidden layer.

**Masked Multi-Head Attention (MMHA)**

The purpose of MMHA is to teach the model how to produce sentences, it masks the target of prediction, the model generates predictions, and said generated predictions are compared to the target.

The masking operation is done through assigning -inf value to masked predictions.

Table

Description automatically generated

The transformer’s decoder uses its output embeddings in train and feeds it to the non-masked multihead attention in the decoder block as query matrix, while the key and value matrices come from the encoder.

This technique was used as it enhances general performance.

**Types of transformers**

There are two types of transformers that are most widely used in research and industry as of currently, one being autoencoding transformers, and the other being autoregressive transformers.

**Autoencoding Transformers**

Autoencoding transformers are the category of transformers that are constituted using only decoder blocks, such as the Bidirectional Representation From Transformers (BERT) model and its variants.

This category of models is, generally speaking, used for discriminative tasks such as sentiment analysis, text classification, and named entity recognition, but can be also used for some generative tasks such as translation and question answering.

**BERT**

The Bidirectional Representation From Transformers [[3](#ref3)], or BERT for short, is a autoencoding transformer that introduced a the pretrain-finetune framework for language modelling (LM) which caused a paradigm shift in NLP.

As an autoencoding transformer, BERT has only the encoder part of the transformer.

The pre-training phase is the stage where BERT is trained on two general LM tasks, Masked Language Modelling (MLM), and Next Sentences Prediction (NSP).

In MLM, a certain percentage of the tokens are replaced with a mask token and the model is to predict them.

In NSP, as the name implies, the model is to predict the next sentence given its predecessor.

BERT at the time of its release achieved SOTA on all benchmarks.

**ELECTRA**

The model ELECTRA “Efficiently Learning an Encoder that Classifies Token Replacements Accurately.” [[4](#ref4)], is model that uses a GAN like approach to make a better use of the training data, the author see the MLM as wasteful as it only uses 15% of the training data,

Their approach was to train a small MLM model, and teach the model to discriminate whether the input token is replaced or original.

The model is competitive with other large language models, but the comparisons done by the authors are not enough to claim SOTA results.

**ALBERT**

The ALBERT model [[5](#ref5)] as the name implies, is A Lite BERT, ALBERT is not only smaller and faster than BERT, but it is also more accurate.

ALBERT has three distinguishing features:

* The removal of Next sentence prediction and using sentence order prediction.
* Factorized embedding parameterization, where the embedding linear layer are decomposed into two small layers, this decreases the memory required to store and use the model, while preserving the performance.
* Cross layer parameter sharing, where the entire model shares the parameters of single block, in practice, the model implements a single encoder block and uses a for loop to iterate the data on it for the number of required encoder blocks.

Just like its predecessor, ALBERT achieves the SOTA on all benchmarks, all while being much smaller, for context, ALBERT X-large is 55% smaller than BERT base while achieving higher performance, and ALBERT XX-large is 70.3% smaller than BERT large while achieving SOTA.

**Autoregressive Transformers**

On the other hand, autoregressive transformers are those that have decoder blocks alongside the encoder ones, such as BART, T5, GPT, etc. and their variants.

This category of models is used for generative tasks such as translation, summarization, question answering, and text generation.

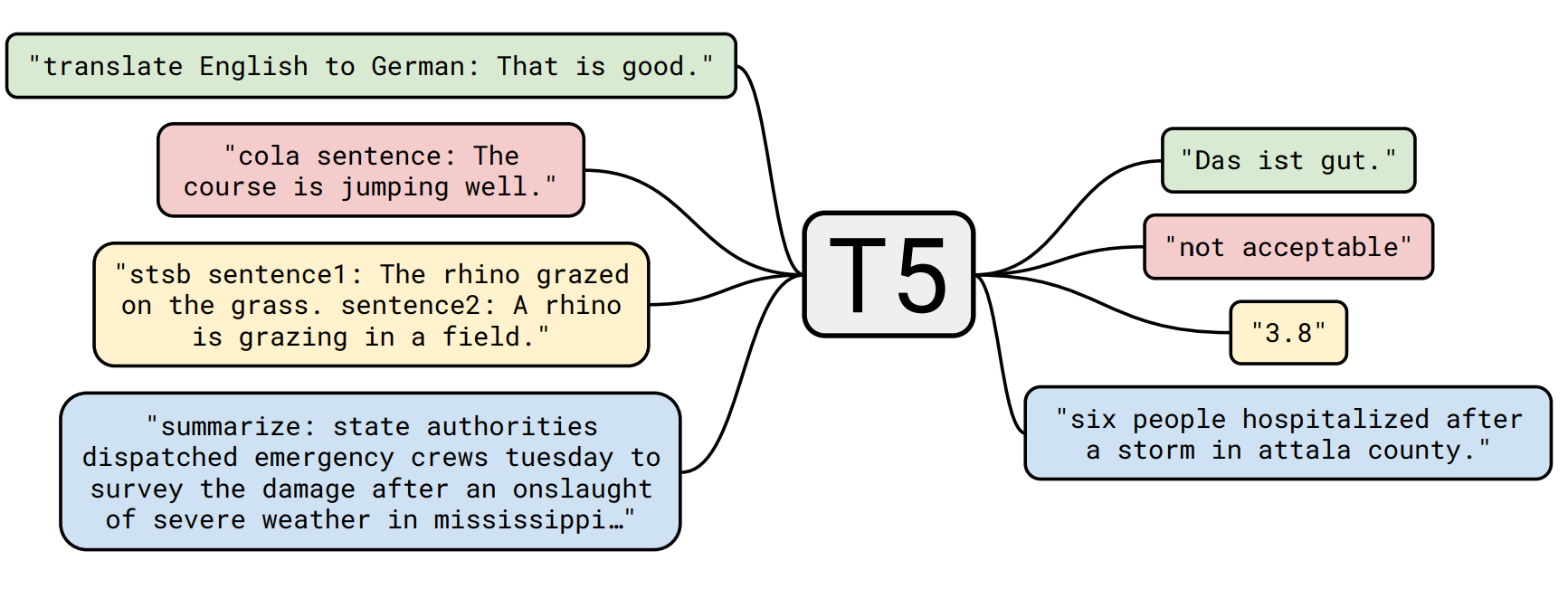
**mT5**

The Multilingual Text to Text Transfer Transformer, or mT5 [[6](#ref6), [7](#ref7)] for short, is a multilingual transformer-based model that the formulates all NLP task as either question answering, language modeling, or span extraction tasks.

The value of such radical method is that it “allows us to directly apply the same model, objective, training procedure, and decoding process to every task we consider.”

The authors exploit their unique methodology to establish a level platform for evaluating and studying the effectiveness of the model at a given task.

The model itself very closely resembles the original transformer with minor modifications.



**BART**

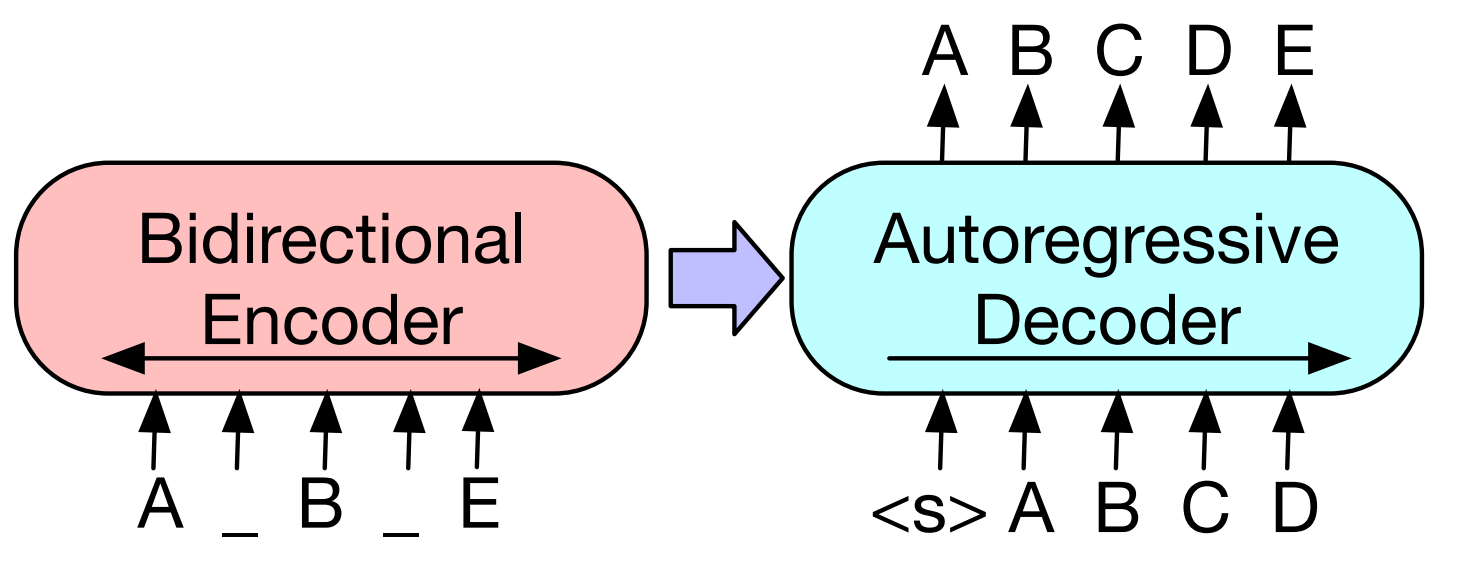
Bidirectional and Auto-Regressive Transformers, a.k.a. BART [[8](#ref8)] is a denoising transformer-based autoencoder that corrupts a document with a noise function, and maps said document to its original.

It is implemented as a sequence-to-sequence model with a bidirectional encoder (BERT like) over corrupted text and a left-to-right autoregressive decoder (GPT like).

The Authors’ motivation was to combine the best properties of BERT and GPT models, since BERT cannot easily generate and GPT cannot learn bidirectionally, combining both of them should solve the issue.



BERT style encoder GPT style decoder



BART

BART is pre-trained by noising it using five different methods:

* Token Masking, as done with BERT, where a certain percentage of the tokens are randomly masked, and the model is to predict the masked tokens.
* Token Deletion, a certain percentage of the tokens is to be deleted, unlike token masking, the model is to not only predict the tokens themselves, but their positions as well.
* Text infilling, A number of text spans (words or phrases) are sampled, with span lengths drawn from a Poisson distribution (λ = 3). Each span is masked using a single masking token, with zero-length spans being represented as masking token inserted into the document.
* Sentence Permutation, where the document is divided into sentences based on full stops, after which said sentences are shuffled in a random fashion.
* Document Rotation, where a token is chosen at random, and the document is “rotated” so that it begins with that token. This task trains the model to identify the start of the document.

Diagram, schematic

Description automatically generated

BART pretraining methods

Such methods of pretraining produce a model that is robust to linguistic noise and allows the model to deeply extract the patterns of the input data.

BART achieves SOTA on Text Summarization on XSum and CNN/Daily Mail, while achieving performance that is comparable to RoBERTa on discriminative tasks.

**Comparison Between mT5 and mBART50**

It is clear that the comparison here is between the mT5 and mBART50 models.

Fortunately, [Sebastian Gehrman *et al.*](https://arxiv.org/pdf/2102.01672v3.pdf) [[9](#ref9)] had done such a comparison and have compared numerous models on different summarization datasets other neural language generation tasks,

The datasets of interest are WikiLingua and MLSum, and both are large multilingual summarization dataset, thus it is reasonable to assume that the performance of the models on them is likely to correspond to a similar performance on the Arabic language, especially since WikiLingua has an Arabic portion.

Note that the WikiLingua dataset is inputted as various languages several times, but all are outputted as English, this is due to the fact the WikiLingua dataset is a cross-lingual abstractive summarization dataset,

i.e., the dataset is designed to allow abstractive summarization to be done such that the input article can be inputted as one language and the summary be outputted as another one.

Looking upon the metrics in the comparison, a pattern emerges, the mBART+ model almost always outperforms the mT5 model except a few times, where it is in its XL from, and mostly by a small margin.

It is thus concluded that the best possible ready model is mBART+, unfortunately the authors do not provide the pretrained mBART+, hence we will operate with the normal [mBART](https://arxiv.org/pdf/2001.08210v2.pdf).

Table

Description automatically generated

**Methods Worth Mentioning**

This section discusses models that are worth mentioning but were not available to use, for any number of reasons.

**PEGASUS [**[**12**](#ref12)**]**

It is a model that is based on BART, but trained in a very unique way which involves masking entire sentences and teaching the model to predict them, this novel idea of pretrain allowed the model to achieve SOTA on 12 benchmark abstractive datasets, of which are XSum, CNN/DailyMail, arXiv & PubMed which are scientific publications datasets.

**Text Summarization using Graph Neural Network [**[**15**](#ref15)**]**

Recently, researchers have combined the Graph-based and Deep Learning-based approaches to extractive text summarization through using a graph attention layer combined with BERT and a supporting network for discovering sub-topics and latent topics within the context of the text, achieving a SOTA record on CNN/DailyMail and NYT datasets.

**Text Summarization using Deep Reinforcement Learning**

Researchers in the past decade have been experimenting with using Deep Reinforcement Learning for text summarization due to its very human-like nature as task, but major success has not been achieved yet.

It is worth noting that GPT-3.5, the model behind ChatGPT uses Deep Reinforcement Learning to deliver human like responses.

**Arabic Specific Models**

Arab researchers have produced Arabic versions of some models, such AraBERT, AraBART, AraGPT2 and more.

**AraGPT2**

AraGPT2 [[11](#ref11)] is a stacked transformer-decoder model trained using the causal language modeling objective, it is trained on 77GB of Arabic text.

AraGPT2 comes in four variants, with the smallest model, base, having the same size as AraBERT-base, rendering it accessible for the larger part of researchers, while on the other hand the larger model variants (medium, large, X-Large) offer improved performance but are computationally more costly and are often more difficult to fine tune.

The AraGPT2 - detector is based on the pre-trained AraELECTRA model fine-tuned on the synthetically generated dataset.

**AraBART**

AraBART [[14](#ref14)] follows the architecture of BART Base (Lewis et al., 2020), having 6 encoder and 6 decoder layers and 768 hidden dimensions.

In total AraBART has 139M parameters, one additional layer-normalization layer was added on top of the encoder and the decoder to stabilize training at FP16 precision. sentence-piece was used to create the vocabulary of AraBART.

The sentence-piece model was trained on a randomly sampled subset of the pretraining corpus, of size 20GB, without any preprocessing. We fixed the vocabulary size to 50K tokens and the character coverage to 99.99% to avoid a high rate of unknown tokens.

**Datasets and Metrics**

The available datasets for the abstractive task:

* [XL-Sum](https://github.com/csebuetnlp/xl-sum): [[18](#ref18)] a multilingual dataset for abstractive text summarization, can be [downloaded](https://huggingface.co/datasets/csebuetnlp/xlsum) from the hugging face library, and the Arabic part can be downloaded directly from GitHub independent form the rest of the dataset,
  + Arabic portion size: 46897.
  + SOTA Model: mT5.
* [WikiLingua](https://github.com/esdurmus/Wikilingua): [[19](#ref19)] A large-scale, multilingual dataset for the evaluation of abstractive summarization systems.
  + Arabic portion size: 29229.
  + SOTA Model: mBART+.

The available datasets for the extractive task, these datasets are quite small compared to the datasets of the abstractive task, and information about them is quite sparse:

* [EASC](https://link.springer.com/article/10.1007/s10579-014-9274-3) (Essex Arabic Summaries Corpus) [[26](#ref26)]: A human generated corpus of summary, best suited to be a test set due to its small size.
  + Article size: 153, summary size: 765
* [KALIMAT](https://sourceforge.net/projects/kalimat/files/kalimat/) [[20](#ref20)] Multipurpose Arabic Corpus: a multi-use dataset that contain generated summaries.
  + Size: 20291

**Metrics**

Possible metrics used for Text Summarization are:

* The ROUGE metrics family.
* The BLEU metrics family, which was developed originally for translation, but it is used for other NLP tasks, as it is a similarity measure.
* The METEOR metric, similar to BLEU, is also a translation metric that can be used for other purposes due to it being a similarity measure.
* The [BLEURT](https://github.com/google-research/bleurt), which is a trained evaluation metric for Natural Language Generation. It takes a pair of sentences as input, a *reference,* and a *candidate*, and it returns a score indicating how much the candidate represents the meaning of the reference.
* The [BERTScore](#ref27) [27] which computes a similarity score for each token in the candidate sentence with each token in the reference sentence using the word embeddings from BERT.

A picture containing text, diagram, line, font

Description automatically generated

The matching is greedy and isolated, where each token in the candidate sentence is matched to the most similar token in the reference sentence and vice-versa for calculating Recall and Precision respectively.

Instead of exact matches, token similarity is computed using cosine similarity between embeddings, the IDF importance weighting could be optionally used.

The precision and recall are combined in order to calculate the F1 score.

**Steps**

1. Tokenize both the candidate and reference sentences.
2. Compute embeddings for each token in both sentences using a pre-trained BERT model.
3. Compute token similarity between each token in the candidate sentence with each token in the reference sentence using cosine similarity of their embeddings.
4. Match tokens between the candidate and reference sentences based on their similarity scores.
5. The Precision, Recall, and F1-score are computed, in this context:

Precision is the fraction of tokens in the candidate sentence that are correctly matched with tokens in the reference sentence. It measures how many of the generated tokens are relevant, it is computed as:

Recall is the fraction of tokens in the reference sentence that are correctly matched with tokens in the candidate sentence. It measures how many of the relevant tokens are generated.

And the F1-score is of course their harmonic mean.

**Our Path**

We started by exploring the different methods that could be used to perform summarization, and hence the options that presented themselves to us were,

1. Recurrent Neural Networks.
2. Rhetorical Structure Theory. [[16](#ref16)]
3. Transformer-based solutions.

We chose to use transformers for a variety of reasons, of which is the beforehand established supremacy of the transformers over the other two options as detailed in the previous comparison we performed.

The transformer is also very widely supported and used which facilities infrastructure and pretrained models.

The GPU Friendly nature was also a contributing factor to our choice, due to the limited nature of the available hardware.

After that we were presented by choosing the sub tasks to work on, we choose the abstractive and extractive subtasks because they are the most broadly studied subtasks.

We searched for a good system design for the extractive task, and we adapted a pipeline that achieved good performance for summarization.

A diagram of a process

Description automatically generated

For the abstractive task, we were faced with the options of using the mT5 and mBART50, and as a result we performed a comparison between to find the best performing model, Fortunately, a previous paper, performed such a comparison and mBART+ a top, which is not open source, so we decided to use mBART50.

Then we found out that there exists an Arabic version of the BART model called the AraBART, which we are currently using.

For the extractive task, previous work utilised the AraBERT model, either through using it alone or alongside clustering.

And as logical extension to that work, we tried to do the same but with improved techniques, such as using better variants of AraBERT [10], such as AraELECTRA [13] and Arabic ALBERT, and utilizing better clustering techniques.

However, doing so proved to be near impossible for a multitude of reasons.

1. All Traditional techniques, not just clustering, are extremely difficult to implement with the Arabic languages, for examples, the very structure of the language is very unique and different from other languages, omitting the option of using methods that rely on the structural properties of the language.

Moreover, the expressive richness of the Arabic language is a large roadblock as well, due to the fact a word could have numerous synonyms, and a meaning could be described many words, making it very difficult to use cluster words using unsupervised and rule-based methods, not to mention the fact that the quality of the summary is very sensitive to the number of clusters.

Another issue is that lingual separators are rarely used in the Arabic language, which makes clustering a nightmare.

1. All autoencoding models were unavailable to train on summarization, they were not implemented on hugging face for extractive summarization, the configuration of hugging face made it impractical for us to implement it ourselves (would have disrupted the entire workflow of the hugging face system and would have made debugging very messy).

Not to mention the fact all available models on GitHub were either already finetuned on other tasks or would have needed the aid of traditional methods.

1. The available data is simply not good enough, KALIMAT, the larger extractive data set is a generated, using it for training kept the model at square 0, figuratively and literary, when training on KALIMAT the produced rouge metrics were all zeros for all epochs.

As a last effort, we tried to fine tune the autoregressive models mT5 and AraBART on EASC and used 50 samples out of the 150 as a validation.

The following are the results from both of them,

Text:

تضع نصب أعينها برامج التجسس يقوم مستخدمو قريبًا بدفعها لشركة للحفاظ أجهزة الكمبيوتر الشخصية خالية برامج التجسس الاستحواذ شركة لمكافحة برامج التجسس إنها ستصدر قريبًا مجموعة أدوات تجرد الأجهزة البرامج المزعجة الرغم كونها مجانية البداية إلا تستبعد فرض رسوم الأشخاص يرغبون تحديث مجموعة الأدوات تظهر الاستطلاعات جهاز كمبيوتر يعمل بنظام تقريبًا موبوء ببرامج التجسس تقوم بكل شيء بدءًا قصف المستخدمين بالإعلانات لسرقة بيانات تسجيل الدخول شركة إصدارًا تجريبيًا مجموعة الأدوات لتنظيف أجهزة يجب متاحًا غضون يومًا تصميم الأداة المساعدة لأجهزة الكمبيوتر تعمل بنظامي التشغيل وستقوم بتنظيف برامج التجسس ومراقبة يحدث جهاز الكمبيوتر باستمرار وسيتم تحديثها بانتظام للتعرف أحدث المتغيرات منح العديد برامج تعزيز الأمان الأخرى جدار الحماية مجانًا مايك ناش نائب الرئيس وحدة الأعمال الأمنية بشركة مايكروسوفت الشركة تزال تعمل حل مسائل التسعير والترخيص تحصيل رسوم الإصدارات المستقبلية يتم خصمه سنضع خطة ونطرحها الخطة مربحة لمايكروسوفت استطلاع حديث أجراه ٪ أجهزة الكمبيوتر موبوءة بالبرامج الخفية يحتوي المتوسط ​​على برنامج تجسس منفصل حاليًا تحول المستخدمون يريدون الحماية برامج التجسس برامج مجانية تأتي برامج التجسس بأشكال عديدة أحسن حالاتها تستغل عادات التصفح البطيئة لتثبيت نفسها وإخضاع المستخدمين لإعلانات مرغوب تخطف النماذج الأخرى إعدادات متصفح الإنترنت لإجبار الأشخاص عرض الصفحات يقوموا بزيارتها الإطلاق تراقب برامج التجسس أكثر حالاتها ضررًا يفعله الأشخاص باستخدام أجهزة الكمبيوتر الخاصة بهم وتسرق معلومات تسجيل الدخول والبيانات الشخصية الأخرى يأتي إعلان مايكروسوفت برامج التجسس اشترت شركة البرمجيات الصغيرة نيويورك يتم الكشف شروط الاستحواذ

mT5:

تسمح شركة مايكروسوفت برامج التجسس يقوم مستخدمو قريبا بدفعها لشركة للحفاظ على أجهزة الكمبيوتر الشخصية خالية برامج التجسس الاستحواذ شركة لمكافحة برامج التجسس اسمها:تقوم شركة مايكروسوفت بمحاولة دمج التقنيتين معاا للاستفادة منهما على مواقع لمكافحة البرامج البرامجالبرامجالبرامجالبرامج المزعجة اهمها اهمها برنامجين يتعلقاعطيات جهازين رئيسيين يتعلقان لتطويرها وتفعيلها شركة للاجهزة اللوحية لمكافحة برامج البرمجيات البرامج البرامج البرامج الخارقة.تتم قراءة و رصدها اثناء قدوما تساحداثها ضعف الحجم الطبيعياظهر فيه

[{'rouge-1': {'r': 0.66, 'p': 0.8, 'f': 0.727},

'rouge-2': {'r': 0.477, p': 0.627, 'f': 0.543},

'rouge-l': {'r': 0.416, 'p': 0.5, 'f': 0.454}}

AraBART:

شركة مايكروسوفت تصدر قريبًا مجموعة أدوات تجرد الأجهزة البرامج المزعجة الرغم كونها مجانية البداية إلا تستبعد فرض رسوم الأشخاص يرغبون تحديث مجموعة الأدوات تظهر الاستطلاعات جهاز كمبيوتر يعمل بنظام تقريبًا موبوء ببرامج التجسس تقوم بكل شيء بدءًا قصف المستخدمين بالإعلانات لسرقة بيانات تسجيل الدخول شركة إصدارًا تجريبيًا مجموعة الأدوات لتنظيف أجهزة يجب متاحًا غضون يومًا تصميم الأداة المساعدة لأجهزة الكمبيوتر تعمل بنظامي التشغيل وستقوم بتنظيف برامج التجسس ومراقبة يحدث جهاز الكمبيوتر باستمرار وسيتم تحديثها بانتظام للتعرف أحدث المتغيرات منح العديد برامج تعزيز الأمان الأخرى جدار الحماية مجانًا مايك

{'rouge-1': {'r': 0.156, 'p': 0.288, 'f': 0.203},

'rouge-2': {'r': 0.0797, 'p': 0.145, 'f': 0.10285713828179613},

'rouge-l': {'r': 0.146, 'p': 0.269, 'f': 0.189}}]

**Results**we have applied the BERTScore metric as it is currently the best metric for text generation.

The following is the distribution of The BERTScore on our test set between the predicted summary and the ground truth for the abstractive model,

A blue and white graph

Description automatically generated

Mean: 0.7709

Variance: 0.0024

Standard Deviation: 0.0489

The high mean performance and low variance signify the performance and consistency of our model on a difficult, general domain test set consisting of 3663 samples.

To get the information loss between the data and the ground truth summary we computed their BERTScore.

A graph of a short score

Description automatically generated

Mean: 0.6705

Variance: 0.0005

Standard Deviation: 0.0223

As for the information loss between the data and the predicted summary, we computed the BERTScore between them as well,

A blue and white graph

Description automatically generated

Mean: 0.7178

Variance: 0.0007

Standard Deviation: 0.0264

Comparing both computational loss between the both of them, it would be safe to assume that our model, given its higher mean performance and almost identical variance and variance, it is safe to assume that our model preserves information properly, and in combination with the BERTScore between the true and predicted summary, we can say that the model is a powerful and consistent summarization tool.

**Us vs ChatGPT**

We have a few advantages over ChatGPT, our model on average takes 3-4 seconds to generate a summary, while ChatGPT takes between 30-40 seconds,

Secondly, our model is much smaller than ChatGPT, it could be store on a single drive and used normal on a single GPU, while ChatGPT is a massive 175,000,000,000 parameter model that needs servers to be used, is not open source, and does not always produce a summary, in the context of it producing a text that is always almost the same size of original text, such that if the article is 100 words, it produces a summary that is 90 words.

While it is true that ChatGPT does produce a summary that has a higher quality, our summary is not significantly worse.

\*Note: this comparison was done on ChatGPT 3.5

**Samples**

Short 20->40, Long 35->65, Extra-long 50->90

1-Text:

يستعد الأهلي للدفاع عن لقبيه المحلي والافريقي وأضاف الأهلي بموقعه على الانترنت أن لجنة الكرة في النادي قررت تجديد التعاقد مع الجهاز الفني للفريق الأول بكامل تشكيله بداية من موسم 2013-2014. وأكدت لجنة الكرة أن تجديد التعاقد في هذا الوقت يأتي دعما لاستقرار الفريق. وكان يوسف قد تولى تدريب الأهلي حامل لقب الدوري المصري ودوري أبطال افريقيا الشهر الماضي خلفا لحسام البدري الذي انتقل للعمل مع أهلي طرابلس الليبي. ويستعد الأهلي لمنافسات الدورة الرباعية النهائية لتحديد بطل الدوري الممتاز مطلع الشهر المقبل. مواضيع قد تهمك نهاية وكانت هذه الدورة قد تاجلت بسبب الأوضاع السياسية والأمنية في البلاد حيث من المتوقع خروج مظاهرات حاشدة ضد الرئيس محمد مرسي في الثلاثين من يونيو/حزيران الجاري. كما سيدافع الأهلي عن لقبه الافريقي في دوري المجموعات حيث يلعب مع غريمه التقليدي الزمالك و ليوبار الكونغولي وأورلاندو بايرتس بطل جنوب إفريقيا

.

Summary:

أعلنت إدارة النادي الأهلي المصري تجديد عقد المدرب محمد يوسف ليقود الفريق الأول بالنادي الموسم القادم.

Short Summary:

قال النادي الأهلي المصري إنه قرر تجديد تعاقده مع محمد يوسف مدرب الفريق الأول لكرة القدم حتى يونيوحزيران المقبل.

Long Summary:

قال النادي الأهلي المصري إنه قرر تجديد تعاقده مع محمد يوسف مدرب الفريق الأول لكرة القدم حتى يونيوحزيران المقبل، وذلك بعد مرور عام على تولي يوسف تدريب الفريق.

Extra-long:

قال النادي الأهلي المصري إنه قرر تجديد تعاقده مع محمد يوسف مدرب الفريق الأول لكرة القدم حتى يونيوحزيران المقبل، وذلك بعد مرور عام ونصف على تولي يوسف تدريب الفريق الأول في مصر خلفا للبرتغالي مانويل جوزيه الذي استقال من منصبه.

summary GPT:

يستعد النادي الأهلي للدفاع عن لقبيه في الدوري المحلي ودوري أبطال إفريقيا، وأعلن النادي عن تجديد تعاقده مع الجهاز الفني بكامل تشكيله للموسم القادم، وذلك بهدف دعم استقرار الفريق. تولى محمد يوسف تدريب الفريق خلفا لحسام البدري، الذي انتقل للعمل مع نادي أهلي طرابلس الليبي. ويستعد الأهلي للمشاركة في الدورة الرباعية النهائية لتحديد بطل الدوري الممتاز، والتي تم تأجيلها بسبب الأوضاع السياسية والأمنية في البلاد، ومن المتوقع حدوث مظاهرات ضد الرئيس محمد مرسي في الثلاثين من يونيو/حزيران الجاري. وسيدافع النادي الأهلي عن لقبه الإفريقي في دوري المجموعات، حيث سيواجه فرق غريمه التقليدي الزمالك وليوبار الكونغولي وأورلاندو بايرتس بطل جنوب إفريقيا.

2-Text: ﻿

ذكر تقرير إخباري أول من أمس أن شهر مايو المشمس يشهد أكبر عدد من حالات الانتحار. وقال باحثون بريطانيون إن عدد حالات الانتحار يزيد في شهر مايو المشمس ليكونأكثر من أي شهر آخر وهم يعتقدون أن الأمر راجع إلى حالة الطقس. وتقول مجموعة برايوري المتخصصة في بحوث الطب النفسي إن الطقس المشمس الذي عادهما يساعد الناس في التغلب على كآبتهم يعطيهم كذلك " القدرة على اتباع دوافعهم الانتحارية." ويقول المسؤول عن الخدمات الصحية في المجموعة البروفيسور كريس تومسون إن هناك علاقة مباشرة بين سطوع الشمس والانتحار. وأوضحت دراسات أخرى أن مستوى السيروتونين يرتفع حسب كمية أشعة الشمس التي يتعرض لها الشخص. وتبين الإحصائيات أن عدد محاولات الانتحار ارتفع بـنسبة 50 % منذ 1990 وأن معظمهن أقدموا على الانتحار كانوا من الرجال.

Summary:

ذكر تقرير اخباري اول من امس ان شهر مايو المشمس يشهد اكبر عدد من حالات الانتحار. وتقول مجموعة برايوري المتخصصة في بحوث الطب النفسي ان الطقس المشمس الذي عادة ما يساعد الناس في التغلب على كآبتهم يعطيهم كذلك " القدرة على اتباع دوافعهم الانتحارية. واوضحت دراسات اخرى ان مستوى السيروتونين يرتفع حسب كمية اشعة الشمس التي يتعرض لها الشخص.

Short Summary:

كشفت إحصائيات جديدة أن شهر مايو المشمس يشهد أكبر عدد من حالات الانتحار في العالم وهو ما يزيد بنسبة 50 % عن أي شهر آخر.

Long summary:

كشفت إحصائيات جديدة أن شهر مايو المشمس يشهد أكبر عدد من حالات الانتحار في العالم وهو ما يزيد بنسبة 50 % عن أي شهر آخر من العام الماضي حسب إحصائيات وزارة الصحة البريطانية.

Extra Long:

كشفت إحصائيات جديدة أن شهر مايو المشمس يشهد أكبر عدد من حالات الانتحار في العالم وهو ما يزيد بنسبة 50 % عن أي شهر آخر من العام الماضي. فما هي الدوافع وراء إقدام هؤلاء على ذلك؟ وما هي الدوافع التي تدفعهم للانتحار؟

summary GPT:

تقرير إخباري يشير إلى أنه في شهر مايو، يحدث أكبر عدد من حالات الانتحار، ويعتقد باحثون بريطانيون أن هذا يعود إلى حالة الطقس المشمس، وأن الطقس المشمس يمنح الناس القدرة على اتباع دوافعهم الانتحارية. كما يشير المسؤول عن الخدمات الصحية في مجموعة برايوري إلى وجود علاقة مباشرة بين سطوع الشمس والانتحار. وتظهر الإحصائيات أن عدد محاولات الانتحار ارتفع بنسبة 50٪ منذ 1990، وأن معظم الذين يقومون بالانتحار هم من الرجال.

3-Text:

وقال وزير الخارجية الإيراني، جواد ظريف، إن إسرائيل هي الدولة الوحيدة في المنطقة التي لديها برنامج سري للأسلحة النووية، مؤكدا أن الوقت قد حان لكي يفتح الإسرائيليون منشآتهم للمفتشين الدوليين. وكان نتنياهو قد ذكر، في خطاب أمام الجمعية العامة للأمم المتحدة، اسم الشارع الذي يوجد فيه الموقع المزعوم، مطالبا المحققين الدوليين بتفتيشه. وتنفي إيران سعيها لامتلاك أسلحة نووية، وتؤكد أن برنامجها النووي أغراضه سلمية. لكن نتنياهو قال إن الإيرانيين يعتزمون استعمال هذه التجهيزات في الوقت الملائم لإنتاج أسلحة نووية، مضيفا أن "إسرائيل لن تسمح لنظام دعا إلى تدميرنا بتطوير أسلحة نووية، لن تسمح اليوم ولا بعد 10 أعوام ولن تسمح له أبدا". وتابع يقول: "ما تخفيه إيران ستجده إسرائيل". وتعترض إسرائيل على الاتفاق النووي الذي وُقع مع إيران في عام 2015، وأشاد نتنياهو بقرار الرئيس الأمريكي، دونالد ترامب، الانسحاب منه. نتنياهو ذكر اسم الشارع الذي يوجد فيه الموقع النووي المزعوم وأكدت الوكالة الدولية للطاقة الذرية أكثر من مرة أن طهران حافظت على التزاماتها بموجب الاتفاق النووي مع القوى الدولية. وظهر نتنياهو أمام الجمعية العامة للأمم المتحدة يرفع خريطة وصورة لمخزن، وحض المفتشين التابعين للأمم المتحدة على تفتيشه، قائلا: "إنني أكشف اليوم لأول مرة أن إيران لها موقع نووي سري آخر". وقال أيضا إن حزب الله "أنشأ ثلاثة مواقع للصواريخ قرب مطار بيروت". واتهم رئيس الوزراء الإسرائيلي "عملاء إيرانيين" بالتخطيط لهجمات في الولايات المتحدة وأوروبا، وقال إن النظام الإيراني "يقمع شعبه بوحشية" منذ عقود. واتهم إيران بالضلوع في أعمال العنف في العراق وسوريا، وبتسليح حزب الله وحماس في غزة، وإطلاق الصواريخ على السعودية. وانتقد سياسة "التهدئة" الأوروبية تجاه إيران، مشبها إياها بتردد العواصم الأوروبية في الوقوف في وجه أدولف هتلر قبيل اندلاع الحرب العالمية الثانية، متسائلا: "هل حفظ هؤلاء القادة الأوروبيون الدرس من التاريخ؟، هل سيستيقظون يوما؟ ثم أجاب: "نحن في إسرائيل لسنا بحاجة إلى من يوقظنا لأن إيران تهددنا يوميا". -------- --- يمكنكم استلام إشعارات بأهم الموضوعات بعد تحميل أحدث نسخة من تطبيق بي بي سي عربي على هاتفكم المحمول.

Summary:

نفت إيران ما قاله رئيس الوزاء الاسرائيلي، بنيامين نتنياهو، بشأن اكتشاف بلاده ما وصفه "بموقع منشأة تخزين سرية لمواد نووية" في طهران.

Short Summary:

. أعلنت إيران أنها تمتلك "موقعا نوويا سريا" في ضواحي العاصمة الإيرانية طهران

long summary:

أعلنت إيران أنها تمتلك "موقعا نوويا سريا" في ضواحي العاصمة الإيرانية طهران، وذلك بعد يوم واحد من إعلان رئيس الوزراء الإسرائيلي، بنيامين نتنياهو، نيته استعمال هذا الموقع في تطوير أسلحة نووي.

Extra-long:

علنت إيران أنها تمتلك "موقعا نوويا سريا" في ضواحي العاصمة الإيرانية طهران، وذلك بعد يوم واحد من إعلان رئيس الوزراء الإسرائيلي، بنيامين نتنياهو، عن نيته استعمال هذا الموقع في إنتاج أسلحة نووية، حسب الوكالة الدولية للطاقة الذرية التابعة للأمم المتحدة.

Summary GPT: Failed

4-Text:

بعد 17 عاما من الصداقة أردني يبحث عن عائلة صديقه المصري سيد عبد ربه الذي توفى وحيدا في عمان. وناشد السلطات المساعدة في الوصول إلى أهله لتيسير دفنه. برنامج #بي\_بي\_سي\_ترندينغ: يمكنكم مشاهدة الحلقات اليومية من البرنامج الساعة الثالثة بتوقيت غرينيتش، من الإثنين إلى الجمعة، وبإمكانكم أيضا الاطلاع على قصص ترندينغ بالضغط هنا.

Summary:

أردني يبحث عن عائلة صديقه المصري الذي مات وحيدا بعمان.

Short Summary:

بعد 17 عاما من الصداقة أردني يبحث عن عائلة صديقه المصري سيد عبد ربه الذي توفى وحيدا في عمان.

long summary:

بعد 17 عاما من الصداقة أردني يبحث عن عائلة صديقه المصري سيد عبد ربه الذي توفى وحيدا في عمان، ويناشد السلطات المساعدة في الوصول إلى أهله لتيسير دفنه. ويقول الشاب المصري لبي بي سي إن "سيد عبد ربه توفي في الأردن.

Extra-long:

بعد 17 عاما من الصداقة أردني يبحث عن عائلة صديقه المصري سيد عبد ربه الذي توفى وحيدا في عمان، ويناشد السلطات المساعدة في الوصول إلى أهله لتيسير دفنه. ويقول الشاب المصري لبي بي سي إن "سيد عبد ربه توفي في الأردن.

Summary GPT:

يبحث رجل أردني عن عائلة صديقه المصري، سيد عبد ربه، الذي توفي وحيدًا في الأردن بعد 17 عامًا من الصداقة بينهما. يناشد الرجل السلطات للمساعدة في الوصول إلى عائلة الصديق المتوفى لتسهيل دفنه، وذلك في إطار برنامج #بي\_بي\_سي\_ترندينغ. يمكن مشاهدة الحلقات اليومية من البرنامج في الساعة الثالثة بتوقيت غرينيتش من الإثنين إلى الجمعة، ويمكن الاطلاع على قصص ترندينغ بالضغط على الرابط المذكور في الخبر.

5-Text:

قالت صاحبة الخفافيش الأسترالية أليسا هيلي إن الفريق الأسترالي يجب أن يكون حذرًا من كابتن فريق الهندي للسيدات هارمانبريت كاور في بطولة كأس العالم للسيدات T20. ، التي تم اختيارها في فريق العام فى 2018 ICC ،ساعدت Harmanpreet الهند في الوصول إلى الدور نصف النهائي من بطولة ICC Women's World T20 2018 في نوفمبر الماضي.

Summary:

يجب على أستراليا الانتباه إلى كابتن فريق الهند Harmanpreet في بطولة كأس العالم للسيدات T20

Short summary:

قالت الأسترالية أليسا هيلي إن الفريق الأسترالي يجب أن يكون حذرا من كابتن فريق T20I الهندي للسيدات هارمانبريت كاور في بطولة كأس العالم للسيدات.

Long summary:

قالت صاحبة الخفافيش الأسترالية أليسا هيلي إن الفريق الأسترالي يجب أن يكون حذرا من كابتن فريق T20I الهندي للسيدات هارمانبريت كاور في بطولة كأس العالم للسيدات T20.

Extra-long summary:

قالت صاحبة الخفافيش الأسترالية أليسا هيلي إن الفريق الأسترالي يجب أن يكون حذرا من كابتن فريق T20I الهندي للسيدات هارمانبريت كاور في بطولة كأس العالم للسيدات T20 المقرر إقامتها في الهند العام المقبل، وذلك بعد أن ساعدت هارمانبرا كاور الهند في الوصول إلى الدور نصف النهائي.

**Web Interface**

The used framework is flask, due to the fact that it is integrated as PyTorch’s choice framework for deployment.

Our web interface accepts input text as:

* Plain text
* Word Document
* Voice
* Image

The criteria we had for the API is that they had to be free and powerful.

We used Google’s speech-to-text API, as it was the most viable option.

For the image, we used the OCR.space API.

For the document we used the docx API.

**The Interface**

The following are screenshots of each page of the website.

A screenshot of a computer

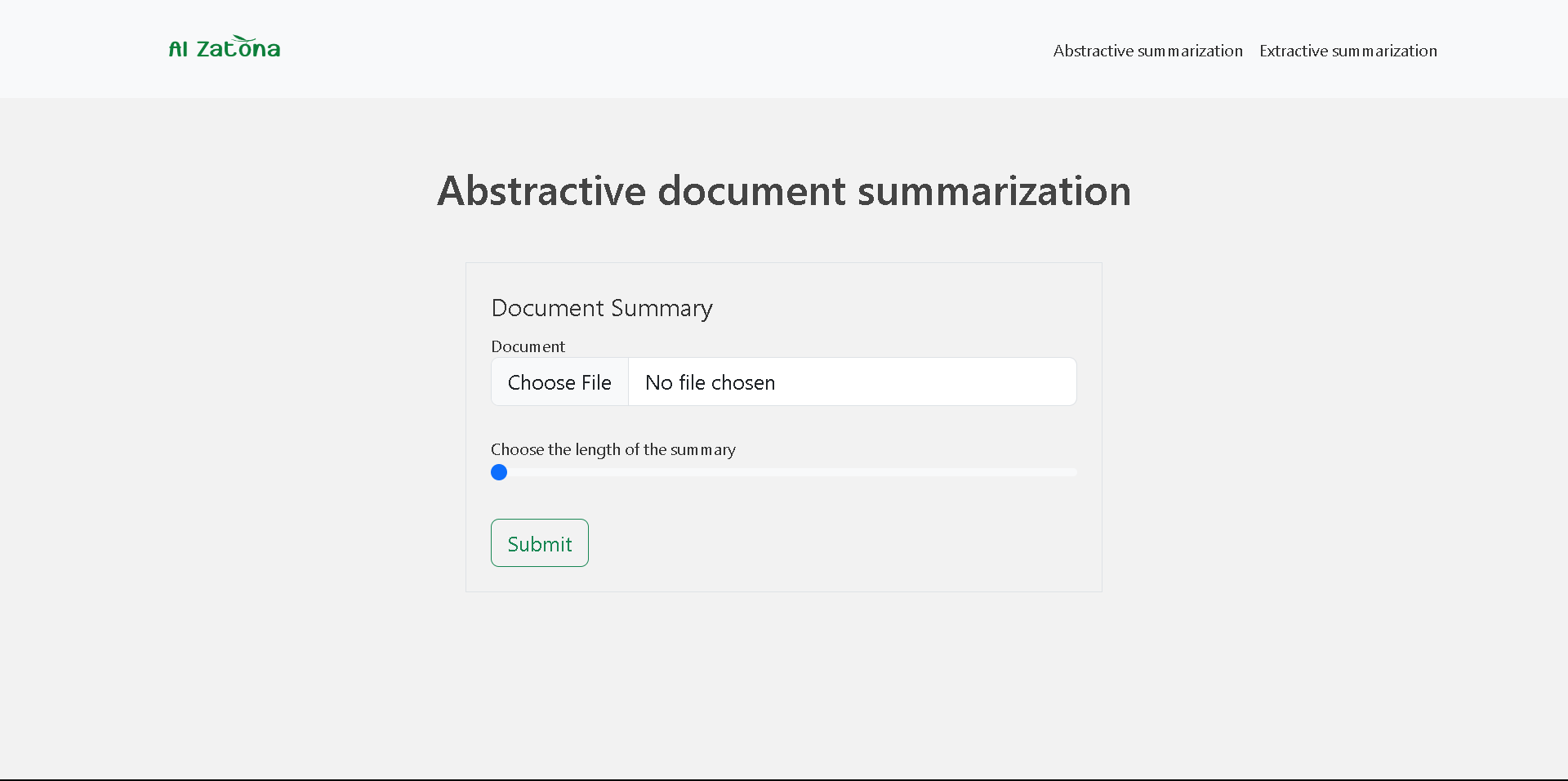
Description automatically generated

The home page, a simple main page laying out the functionality of our system, describing the difference between the two types of summarization as per their definitions, and the button on the top right corners lead to them.

A screenshot of a computer

Description automatically generated

The page for abstractive summarization, presenting the options for input.



Document input, on this page the input is received as a document which is then converted into plain text by the API and is fed to the model.

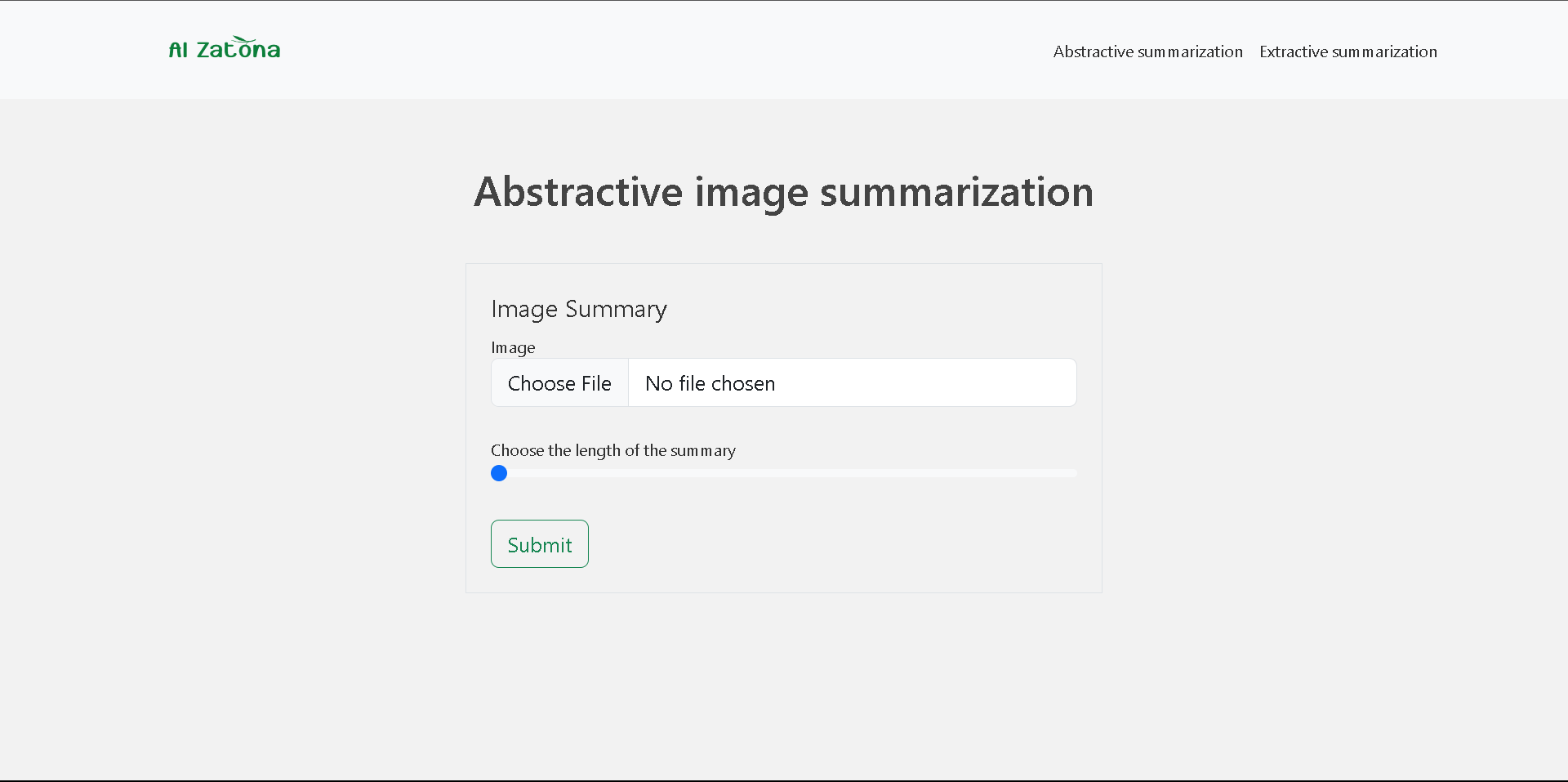


Image input, where an OCR API is applied to retrieve the input text from an image and pass it to the model.

A screenshot of a computer

Description automatically generated

Speech-to-text input, where you as a user upload a voice record representing the input article in audio from, which is then converted to text using the API, and fed to the model.

A screenshot of a computer

Description automatically generated

Plain text, where the input is received from the user is taken direct as text and passed to the model.

A screenshot of a computer

Description automatically generated

All of the previous apply also to the extractive summarization.

**Other Possible Applications**

A large contingent of the workforce is based upon processing large quantities of text, especially in the case of hyper literate profession such law, accounting, banking, finance, and management, all of which spend a copious amount of time read, processing, analysing, and dissecting numerous bodies of text,

The professionals would greatly benefit from having a tool that provide an accurate summary that is faithful to the original source material, as having such summary, would, at the very least aide in facilitating and speeding up the processing of analysing the documents.

Moreover, students of knowledge would benefit from such a tool as well, as the presence of a summary that accurately represents the content of the source material, as this would provide a shortcut for understanding the source material faster, all while providing a revision paragraph for future reference.

It would also aide in speeding up government bureaucracies and matters that take a long time to finish as it would decrease the paper load present.

Summarization tools are not a novel concept, in fact, it is quite frequently, but only in English, and hence providing a summarization tool in Arabic is worthwhile contribution.

**Future Work**

A possible advancement that is somewhat of a “low hanging fruit”, is to use the BERTScore in the loss function of the model, for example the loss function can be,

Theoretically, this alone has the potential to achieve the SOTA, due to the fact that BERTScore correlates highly with human evaluation as demonstrated in its paper, which could potentially mimic the effect of a human reward model used in the RL refinement of GPT 3.5.

Multiple Document Summarization is another track that could be taken, where a compilation of a few documents is processed to produce a single summary, this would be beneficial in producing a report on a series of documents but would require a lot of data scrapping.

Long sequence summarization is good task to work on, as long sequence data is highly available and summaries for it are often desired, the LongNet model has the potential to produce another paradigm shift in AI.

The task that would have the largest impact and would produce the highest value would be the Arabization of the PEAGSUS model, as it is built with abstractive summarization in mind, the dataset would be the dataset used by AraBERT and AraBART

**References**

1. An efficient single document Arabic text summarization using a combination of statistical and semantic features, Qaroush et. al.
2. Attention Is All You Need, Vaswani et. al.
3. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding, Devlin et. al.
4. ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators, Clark et. al.
5. ALBERT: A Lite BERT for Self-supervised Learning of Language Representations, Lan et. al.
6. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer, Raffel et. al.
7. mT5: A Massively Multilingual Pre-trained Text-to-Text Transformer, Kale et. al.
8. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension, Lewis et. al.
9. The GEM Benchmark: Natural Language Generation, its Evaluation and Metrics, Gerhrmann0 et. al.
10. AraBERT: Transformer-based Model for Arabic Language

Understanding, Antoun et. al.

1. ARAGPT2: Pre-Trained Transformer for Arabic Language Generation, Antoun et. al.
2. PEGASUS: Pre-training with Extracted Gap-sentences for Abstractive Summarization, Zhang et. al.
3. ARAELECTRA: Pre-Training Text Discriminators for Arabic Language Understanding, Antoun et. al.
4. AraBART: a Pretrained Arabic Sequence-to-Sequence Model for Abstractive Summarization, Moussa Kamal et. al.
5. Enhancing Extractive Text Summarization with Topic-Aware Graph Neural Networks, Cui et. al.
6. A New Approach for Arabic Text Summarization, Lagrini et. al.
7. Abstractive Arabic Text Summarization Based on Deep Learning, Wazery et. al.
8. XL-Sum: Large-Scale Multilingual Abstractive Summarization for 44 Languages, Hasan et. al.
9. WikiLingua: A New Benchmark Dataset for Cross-Lingual Abstractive Summarization, Faisal Ladhak et. al..

1. [KALIMAT](https://sourceforge.net/projects/kalimat/files/kalimat) Multipurpose Arabic Corpus paper
2. BLEURT: <https://github.com/google-research/bleurt>
3. Decision Transformer: Reinforcement Learning via Sequence Modeling
4. Transformers in Time Series: A Survey
5. Graph Transformer Networks
6. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale
7. Creating language resources for under-resourced languages: methodologies, and experiments with Arabic
8. BERTScore: Evaluating Text Generation with BERT