



uOttawa

Faculté de génie
Faculty of Engineering

DTI5125[EG] DATA SCIENCE APPLICATIONS

BERT EXTRACTION SUMMARIZATION TEXT

FINAL PROJECT

Ekhlas Soliman Khlil mosa

Samah Taher Abdo Ebrahim

Ragab Moawad Ayed Godah

Problem formulation:

Recently people became more busy and they don't have the time to read a whole articles about some topics which they really interested in like sports politics and trending news, and they get board for reading along articles which maybe don't have a valuable information as they expected, so they desperately need a summary of the most important information in the article in order to decide whether the article is interesting enough and they will read the entire article or they will be satisfied with the summary only. Text summarization is playing a vital role to solve this problem and it provides a suitable summary. Text summary is a rundown of the source form of unique information, while keeping its principal substance and helping the user to rapidly comprehend the huge volumes of data. The main idea of summarization is toward removing unusual data from a given content. The synopsis is to discover a subset of information which contains data of whole content. It finds principally the most critical striking and featured words.

Preprocessing:

The most important step in any machine learning model is providing good data to enable the model to learn correctly. If the data has a problem this will affect the performance of the model and lead it to have bad results so preprocessing data is the first step to be done.

The chosen data here is news articles from BBC news, it is a data set containing the whole article and its corresponding summary but in order to enter the article into the model, some text preprocessing step was done

Here at the beginning of the data preprocessing process, data transformed into lower case, then we used regular excretion to remove punctuation and special characters, after tokenization and lemmatization are applied to obtain only the base of the word then stop words are removed

	articles	labels
0	dibaba break world record ethiopia tirunesh di...	sport
1	sullivan commits dublin race sonia sullivan se...	sport
2	colour gardener storm win britain jason garden...	sport
3	isinbayeva claim new world best pole vaulter y...	sport
4	hansen delay return british triple jumper ashi...	sport

Figure 1:subset of data after cleaning and preprocessing step

Methodology:

According to our scenario, we applied 3 classification models and 1 clustering model, and finally the summarization model but after that, we used the feature engineering model and dimension reduction model.

Feature engineering model:

TFIDF, is often used as a weighting factor in information retrieval and text mining, also, it's a numeric measure that is used to score the importance of a word in a document based on how often did it appear in that document, and a given collection of documents, so we use it to vectorize our articles.

Dimension reduction model:

TSNE, it embeds the points from a higher dimension to a lower dimension trying to preserve the neighborhood of that point. So, it's good for NLP tasks.

Classification model:

We used a classification model to predict the new article that the user provide belongs to which label from our 5 labels, and applied 3 model [KNN, AdaBoost, Decision Tree].

AdaBoost, starts by fitting a classifier on the original dataset, then fits further copies of the classifier on the same dataset, but adjusts the weights of poorly classified instances so that succeeding classifiers focus more on difficult cases.

Decision Tree, the goal is to learn simple decision rules from data attributes to develop a model that predicts the value of a target variable. A tree is an approximation to a piecewise constant.

K Nearest Neighbors, which has the highest similarity score among the training corpus, we train the model on entail articles. The categories of these articles are known before the approach is used. In terms of the terms they share, a new document with no category is compared to all of these training documents. Finally, the documents that the new document is most similar to are identified, and the appropriate category is allocated.

Finally, we choice the champion model which gives the highest accuracy [**KNN**].

Clustering model:

we have considered that we don't have labels in data frame and applied clustering model, and the same reason of classification, if the user provide an article and we don't know his label then the clustering can predict the label of that article.

K clustering, is a type of centroid-based clustering algorithm. A centroid is a data point in the center of a cluster (imaginary or actual).

Evaluation:

To evaluate the project, we applied at each model a method to evaluate the performance of it at test the all-performance metric that may tell how will each model will act in the deployment

First for classification models [KNN, AdaBoost, Decision Tree] there are the three- confusion matrix for each one.

Then, for the clustering model which is k-means we used silhouette score which is a metric used to calculate the goodness of a clustering technique. Its value ranges

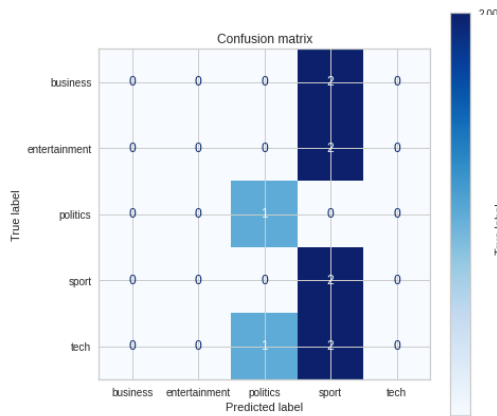


Figure 4:for AdaBoost classifier

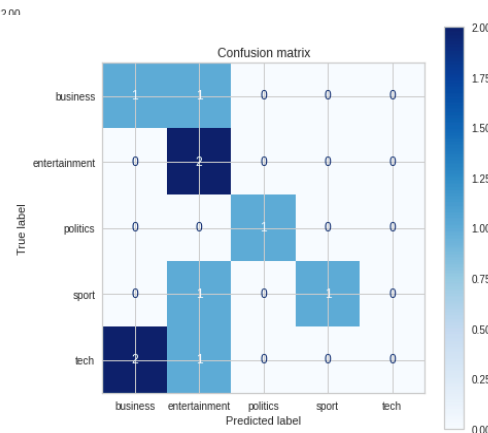


Figure 3: Decision Tree Classifier

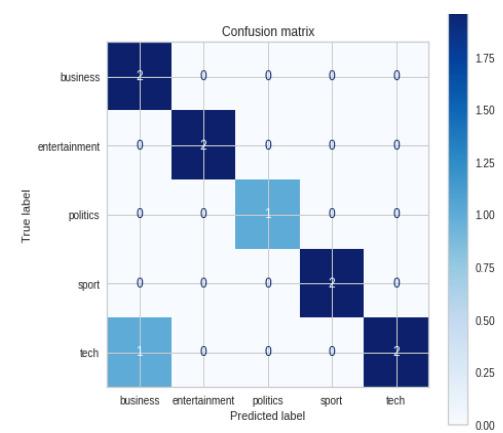


Figure 2: KNN classifier

from -1 to 1. 1: Means clusters are well apart from each other and clearly distinguished. 0: Means clusters are indifferent, or we can say that the distance between clusters is not significant.

-1: Means clusters are assigned in the wrong way.

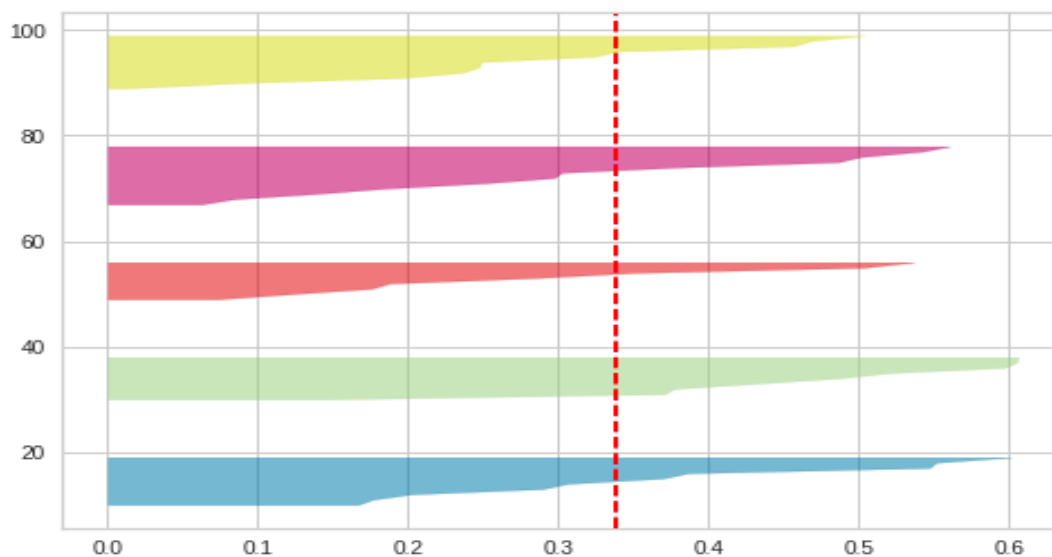


Figure 5:silhouette score for k-means

Finally for the summarization model itself we used the blue method to evaluate the model as it measures the similar between the summaries of systems and the summaries of human

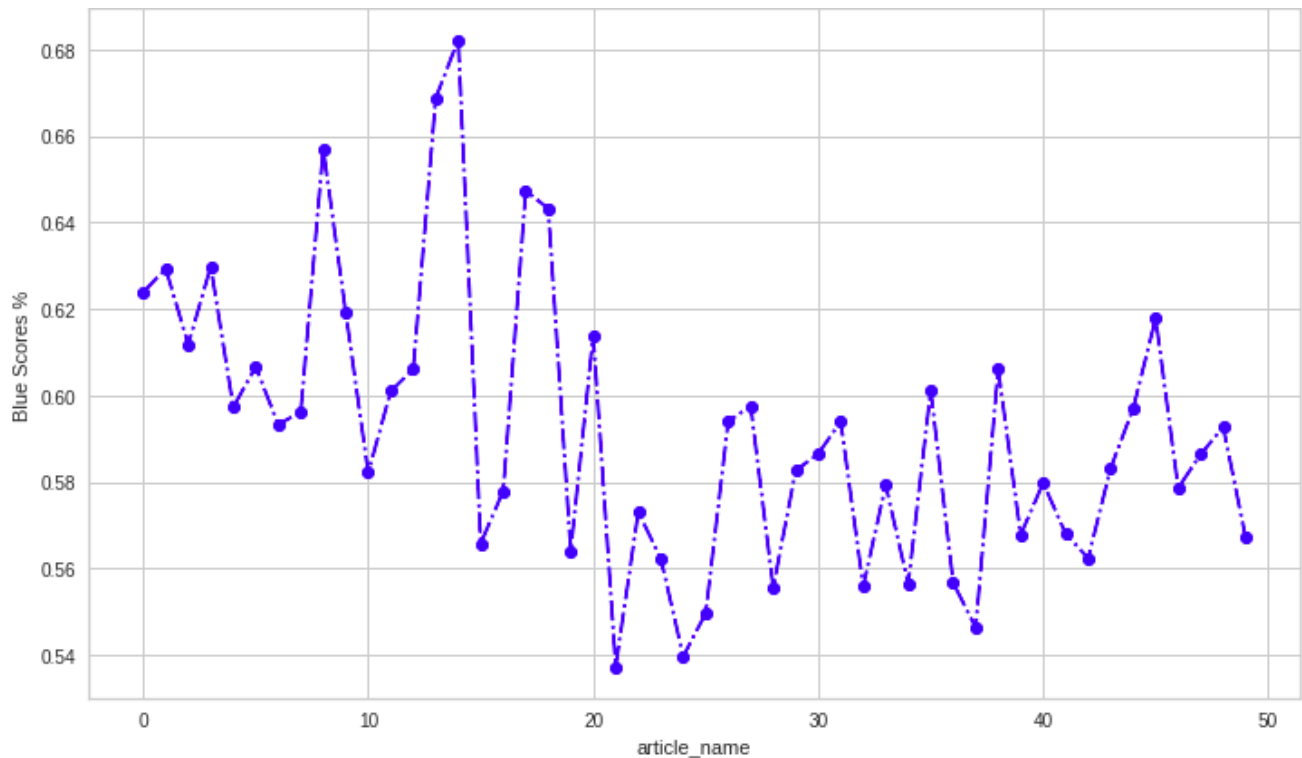


Figure 6:Blue score for Bert summarization model

Steps for the conversation:

- Welcome and greet the user.
- The user could provide or order he wants to summarize a text, then the chatbot will answer and ask about text type (e.g book, novel, article or something like that)

Qs:

1. I want to summarize a text.

I need a text summary.

can you summarize this text for me?

I have a text and I want to summarize it.

I want to make a summary.

And the bot will answer with:

Sure, what kind of text do you have? Or

Great, what type of text do you want to summarize?

2. some articles.

it is part of a document.

some stories.

It is a book.

I have novels.

And the bot will answer with:

great, there are two types to summarize this \$txt_type, what do you want?

**to provide a summary for your \$txt_type, there are two types of
Summarization which one do you prefer?**

3. what is the better among them?

How can I choose?

what is it?

what is the difference?

And the bot will answer with (type of summarization):

4. There if the user responded with these expirations

Then, do with the Extractive type.

great please go on with Extractive.

Extractive is good.

I think Abstractive will be better.

ok, I prefer abstractive

the bot will answer with:

Great, I will summarize with \$user_choose.

\$user_choose! nice, I will go by that.

ok, \$user_choose is a very good choice

Or if he responded with these:

choose the best.

I don't know.

I do not care.

The bot will answer with a PROMPTS response:

sorry, you should choose the type of summarization.

5. how can I send the text?

what's a type of file should I send?

what is the type of file you want?

what's the way you prefer of sending the file?

what's the way you prefer of sending the file?

And the bot will answer with:

you can provide it in just a paragraph or pdf file. what is the maximum length

do you need it?

you can send the link here or the text directly, what is the length you need?

6. one-third of the original one.

less than four sentences.

middle of the paragraph.

half of the original text.

three sentences or lines.

And the chatbot will answer with:

ok, you can send your paragraph.

nice, now send me your text, please.

suitable reduction, send your text now.

Finally, the user can send his article to chatbot like the below links

<https://raw.githubusercontent.com/RagabRadwan/textSummary/main/article01.txt>

<https://raw.githubusercontent.com/RagabRadwan/textSummary/main/142.txt>

and the chatbot will reply with the summery.

Blow in fig (7) and fig (8) screen shoots for greeting and test case scenario:

We used the default greeting intent as it contains several responses for welcoming the user, then the user can type his needs and the bot can understand the request in many formats as shown right

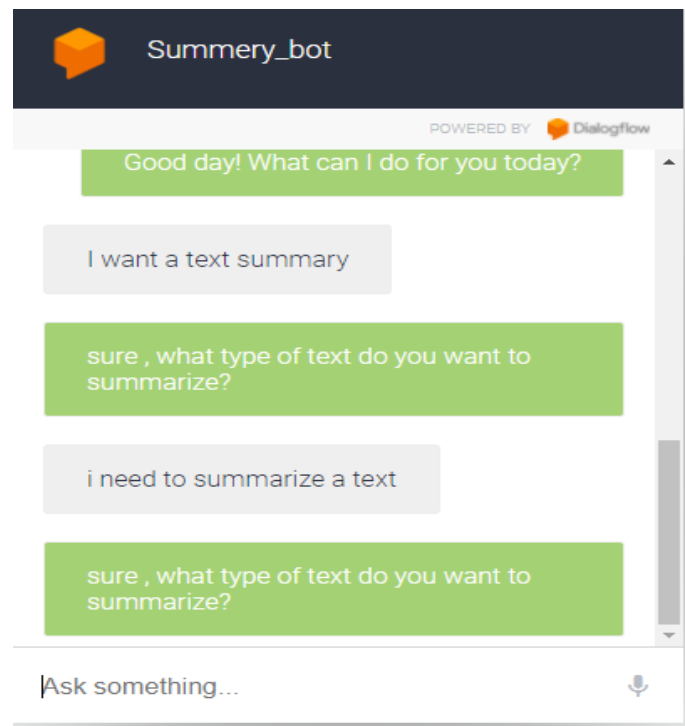
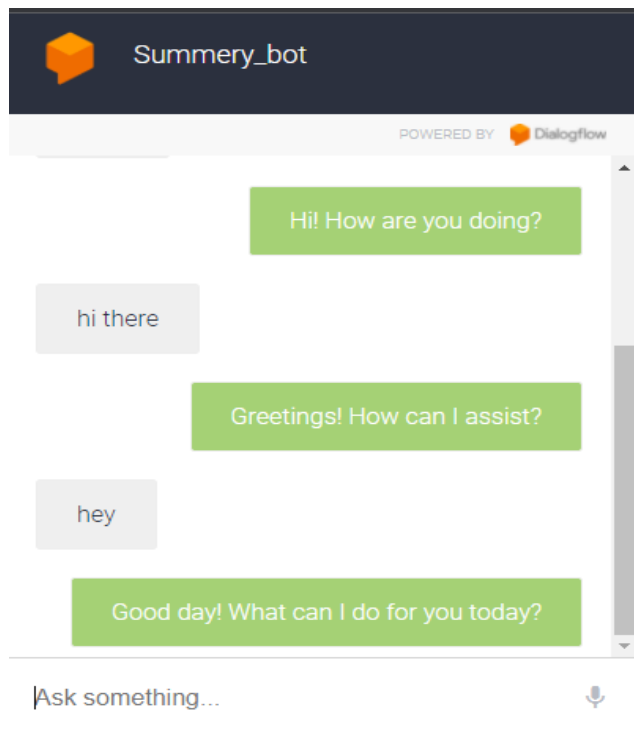


Fig 7: Screen shoots of chatbot welcoming

Then the user will have the flexibility to provide any format of text he wants and also choose the type of text summarization

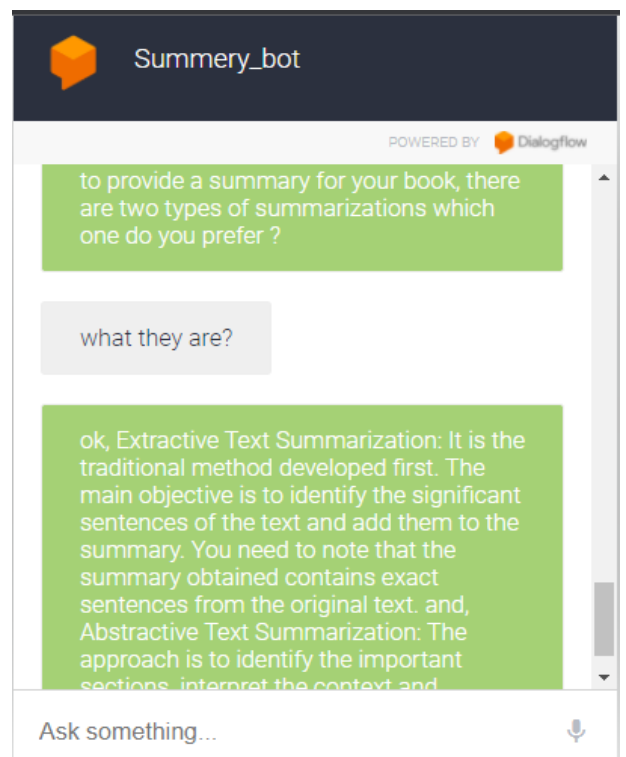
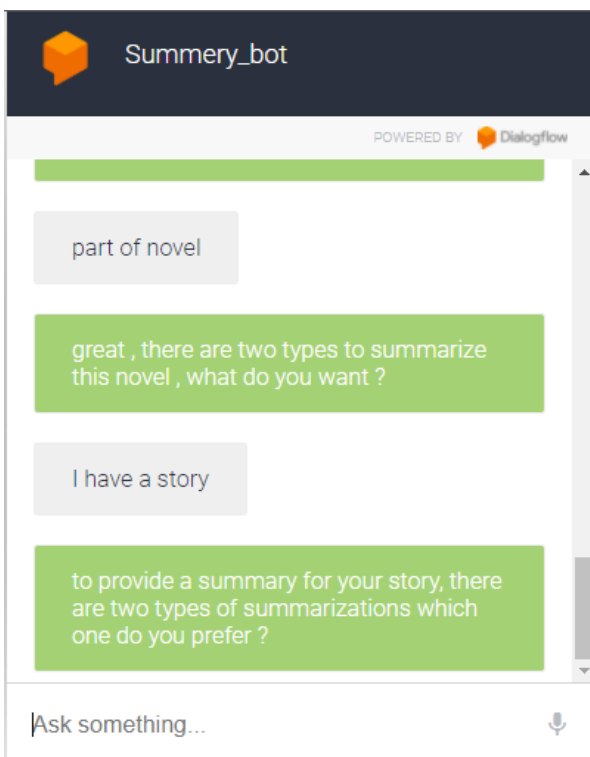


Fig 8: Screen shoots of chatbot testcase

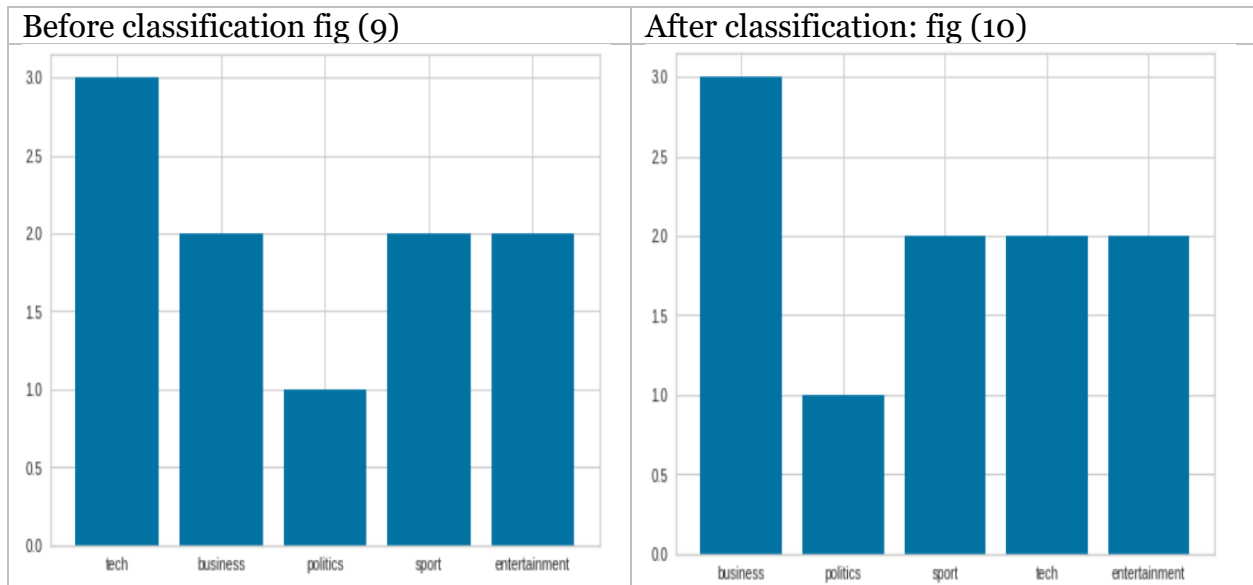
Error Analysis:

To have a deeper understanding of what types of errors our models are making, we did a further error profiling and analysis to the results of the clustering, classification and the BERT summary text and we compared it with the actual classes and summaries of the original dataset. Inspired by the methodology from the different data science topics which we go through in our course, in this context, the error analysis in our project goes through three steps.

KNN classification model:



We applied error analysis in the results of KNN classifier as it achieved the highest accuracy among the other three classifier models. So, by analyzing the test set before and after the classification process.

In figs (9) and (10) Class distribution before and after implementing the classification model:





From the previous bar charts fig (9) and fig (10), we could get some insights about the testing set before and after classification and through these insights we noticed that the misclassification error happens between two main genres business and tech, furthermore to identify the error causes of the misclassifications articles we applied the word cloud frequency before and after classification process to have a deeper understanding about the error causes.

Word cloud before classification.

Fig (11) word cloud of business articles	Fig (12) Word cloud of tech articles
<p>WordCloud of book business</p> 	<p>WordCloud of book tech</p> 

From the above two figs, (11) and (12). We can find some words and topics which may the classification Model be confused in classing some articles as they are related to each other, we may find some word like Apple, Microsoft, google and world which may considered common topics in both genres as they are related to both context and also, they are tending in the articles related to both tech and business.

Word cloud after classification process:

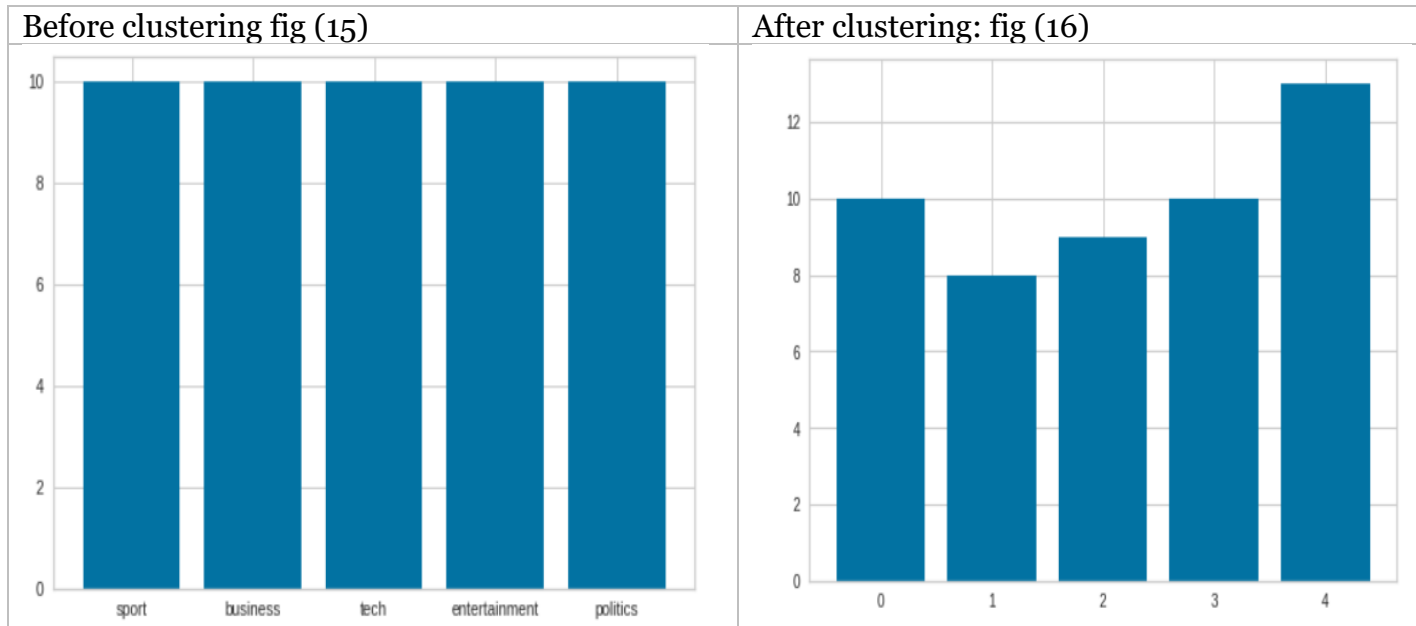
Fig (13) word cloud of business articles	Fig (14) Word cloud of tech articles
<p>WordCloud of book business</p> 	<p>WordCloud of book tech</p> 

We could get some insights from comparing the word cloud of both business before and after classification model implementation. We could notice the absence

of some words in tech genre like spyware trojan and Microsoft, also there is a presence of some words like Sullivan, indica and Sonia, the same absence and presence happened in business genre with different words. We could illustrate these phenomena according to some reasons one of them as the test set isn't large enough due the timing out exception which we received while implementing the pipeline in the article's dataset.

K-Mean clustering Model:

In figs (9) and (10) Classes distribution before and after implementing the clustering model:



From fig (15) which represents the different genres distribution of the whole dataset before applying the clustering model, from the mentioned figure we could notice that the balancing of the different genres in the dataset, but after clustering we found that there is no balancing in the results clusters. Fig (16) illustrates the unbalancing of clusters after clustering by k-means algorithm. Some clusters get more frequency like cluster 4 while other clusters their distribution decreased with a significant numbers like cluster 1 and also cluster 2.

To have a deeper understanding of what types of errors our model is making and causing the fluctuation of clusters distribution we applied the word cloud visualization before and after implementing the cluster model.

Fig (17) word cloud of cluster (4) articles

Fig (18) Word cloud of tech articles

WordCloud of book 4

WordCloud of book tech



By using the word cloud visualization before and after implementing the cluster model. in figs (17) and (18) we could get some insights about the clusters unique words which make us able to match each cluster to its right genre which consider an importing advantage of using word cloud visualization, so it leads us to match cluster 4 to its right genre which is the tech genre, we could also find that the absence of some words after clustering like seeking, spyware, trojan and drive. In the other hand we noticed the presence of some words like share, Christmas, Pernod and takeover.

Fig (19) word cloud of cluster (1) articles

Fig (20) Word cloud of entertainment articles



Furthermore, we applied word cloud visualization for the articles in clutter (1) which achieved a significant decrease in their distribution after clustering model implementation. And matched it with the original genre which has the same unique words which identifying the main characteristics of the entertainment genre, after matching cluster (1) with its right genre, we found there is presence of some new words like prize, west and history as their frequency increased after clustering due to the errors happened in clustering process. Also, we notice the absence of some unique words of entertainment genre like unveils, interactive and Christmas which appeared in the word cloud of cluster (4) in fig (17) and it considered as unique word related to cluster four topics as their frequency in the cluster increased due to the confusing of the model in clustering some articles.

Even though the BERT model produces satisfactory performance for extractive summarization, there is still scope for improvement. and a huge gap which many research papers tried to fill this gap. Hence, we investigate cases further, where even the fine-tuned BERT model goes wrong. We perform this analysis for both actual articles and their summarizations which results of BERT extractive summarization model. To start with, we observe some interesting cases for which the model does not generate the desired summary due to the fact that some information in the article's summary is actually not present in any of the source articles documents. In this context we used word cloud visualization to get deeper understanding of the main characteristics that identify each genre and make its articles unique from the others genres articles.

The above figs (21) and fig (22) indicates the word cloud visualization for articles of sport genre and its summarizations which are results of applying BERT

extractive summarization model. By analyzing the both word clouds. We noticed that the absence of some most frequented words like Sullivan, Sonia, storm and new drug. While there are significant increases in the frequencies of some words like World, record, British, Greene and mass. We could refer to BERT main methodology in those errors as BERT Extractive summarization extract important sentences as they appear in the original document. While BERT identify the importance of some words from their context meaning not from the word frequency in the document or even the meaning of the word as individual and independent word. Furthermore, from figure (6) we could notice that the fluctuation of the Blue scores for BERT summarization model which indicates appearances of some errors in some articles summarizations. But in general BERT summarization considered as the best text summarization model which produce the most accurate summarizations.

Innovativeness and future work:

we are providing a dynamic text summarization system in which the user could use to summary any kind of documents just by passing the document URL through the system chatbot which also considered a valuable automated assistant as it provides the required assistance and guidance through summarization process.

We intend to implement abstractive text summarization as future work, Also, we will provide a service which will let the user choose its preferred topic and it will respond with a summary of the trending articles related to this genre, which will collect those summarizations through web scraping of most famous news websites integrated with social media different platforms trends. Furthermore, the user could use another service through our chatbot by passing certain news articles to our automated chatbot and it will respond to the user indicting whether the news in the article is fake or not.

Conclusion:

In conclusion though out our project we applied three different classification models and we chose the champion model the highest accuracy in classifying the articles each for its proper class genre. Moreover, K-Means clustering model was applied, which we tried to cluster whole articles each in its proper cluster based on its similarity with other articles which belongs to the same cluster. Also, BERT model was applied to get the most accurate summary for given article. Furthermore, We had integrated the proposed model for extractive text summarization to an automated chatbot, through using Dialog flow for implementing the chatbot and Flask frame work as a backend in which our

proposed model is executed and working to service the user requests dynamically and respond with the summarizations for their articles, we integrated both chatbot and the flask backend through Ngrok which provided https public link, which used as a webhook in the chatbot and it invoke a post request with an URL containing the user article and responding with the article summary. In general our chatbot is an automated assistant which provide the required assistance to the Text summarization system users on behalf of the system admin to guide them through out the text summarization concepts which we applied in our project, all of these assistances would provide a better user experience, we also planning to make our chatbot more dynamic and comprehensive to provide more information in details about the text summarization process and we have integrated it with our web page. The users could provide the chatbot with an URL of the article and it will respond with its summary dynamically.