Article

WATS-SMS: A T5-based French Wikipedia Abstractive Text Summarizer for SMS

Jean Louis Ebongue Kedieng Fendji [1]*, Désiré Manuel Taira [2], Marcellin Atemkeng [3],* and Adam Musa Ali [2]

[1] Department of Computer Engineering, University Institute of Technology, University of Ngaoundere, P.O. Box 454, Ngaoundere, Cameroon
[2] Department of Mathematics and Computer Science, Faculty of Science, University of Ngaoundere, P.O. Box 454, Ngaoundere, Cameroon; desiretaira@gmail.com, adammusa2222@gmail.com
[3] Department of Mathematics, Rhodes University, Grahamstown 6140, South Africa
* Correspondence: lfendji@gmail.com (J.L.E.K.F.), m.atemkeng@ru.ac.za (M.A.)

Abstract: Text summarization remains a challenging task in the Natural Language Processing field despite the plethora of applications in enterprises and daily life. One of the common use cases is the summarization of web pages which has the potential to provide an overview of web pages to devices with limited features. In fact, despite the increasing penetration rate of mobile devices in rural areas, the bulk of those devices offer limited features in addition to the fact that these areas are covered with limited connectivity such as the GSM network. Summarizing web pages into SMS becomes, therefore, an important task to provide information to limited devices. This work introduces WATS-SMS, a T5-based French Wikipedia Abstractive Text Summarizer for SMS. It is built through a transfer learning approach. The T5 English pre-trained model is used to generate a French text summarization model by retraining the model on 25,000 Wikipedia pages then compared with different approaches in the literature. The objective is twofold: (1) to check the assumption made in the literature that abstractive models provide better results compared to extractive ones; and (2) to evaluate the performance of our model compared to other existing abstractive models. A score based on ROUGE metrics gave us a value of 52% for articles with length up to 500 characters against 34.2% for transformer-ED and 12.7% for seq-2seq-attention; and a value of 77% for articles with larger size against 37% for transformers-DMCA. Moreover, an architecture including a software SMS-gateway has been developed to allow owners of mobile devices with limited features to send requests and to receive summaries through the GSM network.

Keywords: Text summarization; Fine-tuning; Transformers; SMS; Gateway; French Wikipedia

1. Introduction

One of the most fascinating advances in the field of artificial intelligence is the ability of computers to understand natural language. Natural Language Processing (NLP) is an active field of research encompassing text and speech processing. With the huge amount of textual content available on the Internet, the search for relevant information is becoming difficult. Therefore, one tremendous application of NLP remains the text summarization. According to Maybury [1], text summarization aims to distil the most important information from one or several sources to generate an abridged version of the original document(s) for a particular user or task. Text summarization is usually composed of three steps: pre-processing, processing, and post-processing [2]. The first step prepares the text to summarize by performing some tasks such as sentence segmentation, word tokenization or stop-words removal. The second step really performs the summarization of the text using a summarization technique based on one of the three approaches: extractive, abstractive, and hybrid. Finally, the post-processing aims to address some last issues such as anaphora resolution before generating the final summary [3]–[5].

Text summarization is applied for several purposes including news summarization [6], [7], opinion or sentiment summarization [8], [9], tweet summarization [10], email

summarization [11], scientific papers summarization[12], [13] and even web page summarization [14]. The latter offers the potential to provide an overview of web pages to devices with limited features.

However, although the rate of use of mobile devices is increasing in remote and/or rural areas of developing countries, the bulk of those devices offers limited features with no browser and limited connectivity options [15]. The choice of basic phones with limited features is justified by three main reasons. First, it is the cost. Even if the price of smartphone is decreasing, it is still difficult for people living in rural areas with less than USD 1 per day to afford a smartphone [16]. Second is the lack of energy and the need of long-lasting battery. A full battery recharge of a smartphone usually lasts a day, compared to basic phones that can last several days. The third and not the least is the network coverage. Due to the lack of guarantee of return on investment for Telco operators, most part of the rural regions are unconnected and struggle with simple GSM network in the best case. This is justified by the low Internet penetration rate in developing countries, mainly in sub-Saharan Africa, as shown by the Internet World Stats (IWS)[1]. According to IWS, the average penetration rate of Internet in Africa in the fourth quarter of 2020 is only 43% compared to 64.2% as the world average.

Despite efforts that have been made to provide Wi-Fi-based wireless connectivity in certain locations [17], [18], a large part of those devices is working only with the GSM network. Therefore, the only mean to provide relevant information to those users is through SMS.

An important sector in rural regions, especially in sub-Saharan Africa, remains education that experiences teacher shortage [19], [20]. Providing relevant content to sustain the sector has become a priority. Among freely available educational resources, the Wikipedia encyclopedia represents one of the largest and most recognizable reference resources [21]. It is therefore an excellent choice to mitigate teacher shortage by providing relevant information to students. But due to character limitation when using SMS, a complete Wikipedia page cannot be always sent through SMS, at the risk of losing relevant information. Thus, the summarization of the page is mandatory to provide the quintessence of the page.

Several works have shown potential to summarize Wikipedia pages (e.g. [22], [23]). These works are mostly based on extractive techniques that amount to ranking sentences and selecting the best ones. Usually, they produce summaries that are between one quarter to half the length of the original page.

Although some works such as Lin et al. [24] deals with word limitation when generating the summary, it remains very difficult to consider character limitation as imposed by SMS. Indeed, the summary should fit in a maximum of three SMS. Depending on the mobile carrier, the number of characters should not exceed 455, otherwise the message will be transformed into MMS, requiring adapted devices and additional cost compared to SMS. Moreover, due to the unpredictable relationship between the number of words or sentences and the number of characters, summarizing a webpage into SMS is a real challenge.

The first attempt to consider text summarization with character limitation is provided in [25]. Authors used an extractive approach based on the combination of LSA and TextRank algorithms. Although their approach generates summaries with a limitation on the number of characters, the major constraint is the redundancy while producing the summary, and the lack of context understanding. In fact, the generated summary is based on the computation of the weight of the words in the sentences, with a complete unawareness of the context. Although some other works try to understand the context during the summarization [26]–[31], they do not take into consideration the limitation in terms of the number of characters.

---

[1] https://www.internetworldstats.com/stats1.htm

This paper introduces WATS-SMS, a French Wikipedia Abstractive Text Summarizer that aims to summarize French Wikipedia pages into SMS and to provide summaries directly on the user's device. It is built by applying a transfer learning technique to fine-tune a pre-trained model on French Wikipedia pages. The length of the summary is checked before producing the final summary. The fine-tuning process quickly becomes the fast approach used to build a custom model due to the huge requirements needed to train a model from scratch and the good performance shown by fine-tuned models [32].

The rest of the paper is organized as follows: Section 2 briefly presents the related works on text summarization. Section 3 explains the general architecture of WATS-SMS, as well as details of its main components. Section 4 presents and discusses the quality of summaries based on comparison with existing approaches. Section 5 presents a demonstration of the solution on a basic phone. This paper ends with conclusions and directions for future works, as discussed in section 6.

2. Related works on text summarization

The first significant contributions in text summarization dates back to Luhn [33] in the late 50's and Edmundson [34] in the early 60's. The approaches were basic, focusing on the position of sentences and the frequency of words to produce a summary composed of the extracts from the original text. Since then, many efforts have been made to develop new approaches to text summarisation. Recent surveys such as [2], [35] propose a comprehensive review of the state of art in automatic text summarization techniques and systems. Figure 1, adapted from [2], provides a classification of the automatic text summarization systems.
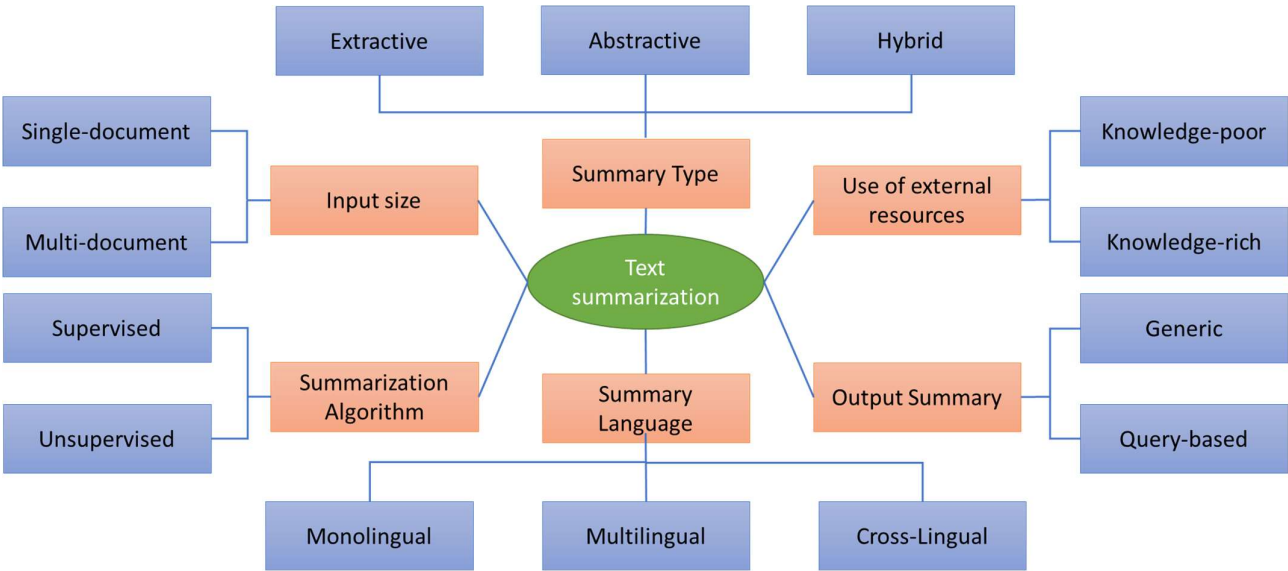

Figure 1. Classification of text summarization systems adapted from [2].

Text summarization can be classified based on the input size. Some approaches such as [36], [37] have been designed for single document while others are targeting multi-document [38], [39]. The summary language is another important aspect in text summarization. Although most of the works are focused on English, some have developed multilingual text summarisation approaches [8], [40], [41], and recently cross-lingual approaches have emerged [42], [43]. Text summarization can be performed based on the type of summary: Extractive, Abstractive, or Hybrid. The first approach consists in scoring sentences and selecting the subset of high-scored sentences [44]. In the second approach, the input document is represented in an intermediate representation and new sentences are generated to build up the summary [45]. The last approach is just a combination of both previous approaches [5].

Abstractive approaches can generate better summaries using words other than those in the original document [47], with the advantage that they can reduce the length of the summary, compared to extractive approaches [48]. Several methods have been developed for abstractive text summarization including rule-based, graph-based, tree-based, ontology-based, semantic-based, template-based, and deep-learning based. The latter was made possible thanks to the success of sequence-to-sequence learning (seq2seq) that uses a set of Recurrent Neural Network (RNN) based on attention encoder-decoder. However, RNN models are slow to train, and they cannot deal with long sequences [49]. Long Short-Term Memory (LSTM [50]) draw their potential from the latter, they are capable of learning long sequences. Several works have reported promising results in a wide variety of application where data has long sequences dependencies.

A breakthrough that revolutionized the NLP industry and helped to deal with long term dependencies and parallelization during training is the concept of transformers introduced in [51]. Later, the BERT (Bidirectional Encoder Representations from Transformers) representation model which is a pre-trained model that considers for the first time the context of a word from both sides (left and right) has been developed in [52]. BERT provides a better understanding of the context in which a word has been used, in addition to allow multi-task learning. BERT can be fine-tuned to generate new models for other languages. More recently, models based on BERT such as CamemBERT [53] and FlauBERT [54] have been trained on very large and heterogeneous French corpus. The generated French models have been applied to various NLP tasks including text classification, paraphrasing, and word sense disambiguation. However, the application of BERT-based model to summarization is not straightforward. In fact, BERT-based models usually output either a class label or a span of the input.

Text summarisation tasks using a BERT-based model are proposed in [55], [56]. Generated models have been evaluated on three datasets of news articles: CNN/DailyMail news highlights, the New York Times Annotated Corpus, and Xsum. Lately, the Text-to-Text transfer transformer (T5) model and framework that achieve state-of-the-art results on several benchmarks including summarization has been proposed [57]. To the best of our knowledge, no work based on T5 deals with the text summarization using French Wikipedia. In addition, the limitation in terms of number of characters has not yet considered during abstractive summarization process.

Due to the homogeneity of Wikipedia in terms of genre and style, and because of the diversity of the domains we are interested in coverage through the WATS-SMS application (news, education, health, etc.), we used the T5 model instead of BERT. T5 works as a ledger of all NLP tasks into a unified format, which is different from BERT-based models that usually generate either a class label or a span of the input [58].

3. WATS-SMS system

This section presents the proposed WATS-SMS system including the SMS mobile gateway and the summarizer module. The complete working of the system is illustrated in Figure 2.

The process starts with the user sending an SMS (through the GSM network) to the SMS gateway (SMS Gateway phone number). The first module of the SMS Gateway, namely the SMS Handler, retrieves the keyword and the phone number of the user. Then the second module of the SMS Gateway, namely the Request Handler, will generate the web request to send to the server. Once the request is received by the summarizer on the server, the Wikidump module will first retrieve and clean the corresponding Wikipedia page to be sent to the Text Summarization module. The latter module generates the summary that takes the reverse path to the user. As mentioned earlier, to prevent the SMS to be transformed into MMS, the number of characters is limited to 455 (the length of three SMS, depending on the Mobile Carrier).

3.1. The SMS Gateway

The SMS Gateway is an application that can be installed on a smartphone to avoid traditional hardware SMS Gateway. It is composed of two modules: The SMS Handler and the Request Handler.
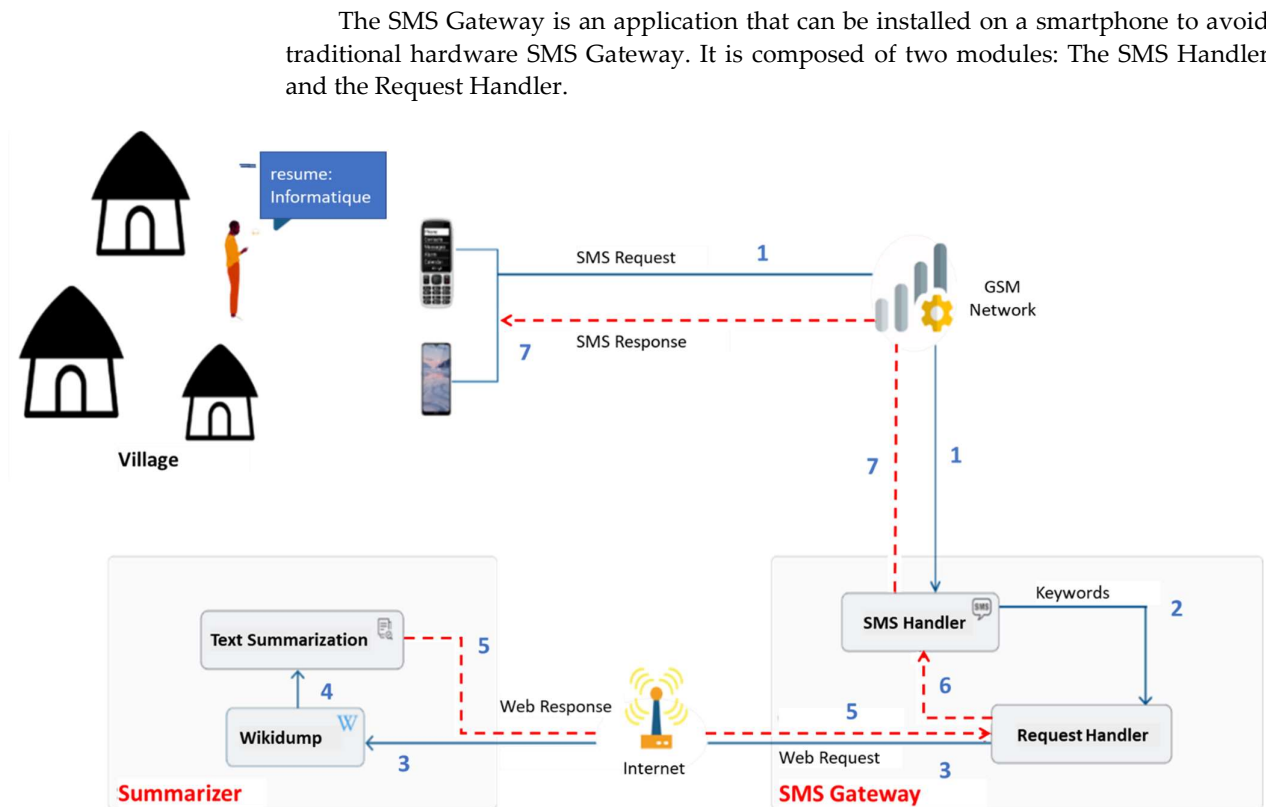

Figure 2. WATS-SMS Architecture.

The SMS Handler performs two functions. Firstly, it reads and filters incoming messages to select those containing the keyword "resume". Since an ordinary smartphone can be used to play this role, not all incoming SMS messages are necessarily requests. Only SMS messages containing the keyword are considered as such. So, the SMS-Reader parses the content, and the text next to the keyword "resume" is considered as the title of the requested Wikipedia page. It can be a simple word such as "informatique" or a set of words such as "corona virus". Two main permissions are required to handle SMS: READ_SMS and SEND_SMS.

The second module of the SMS gateway is the Request Handler which is a JSON parser. It breaks down into words each incoming request to transform it into a JSON format. The result is then sent to the server. This module also works in the reverse way, meaning it converts elements from JSON format into strings. The outgoing JSON file to the server contains the user's phone number and the title of the page. The incoming JSON file from the server contains the user's phone number and the summary of the requested page.

3.2. The Summarizer

The proposed summarization approach consists of three main steps:

1. Pre-processing of the requested page (Retrieving and cleaning);
2. Summarization process;
3. Post-processing the summary.

3.2.1. Pre-processing of French Wikipedia pages

The pre-processing aims to retrieve and clean the content of a requested page before starting the summarization. It is important to know the structure of a Wikipedia page before starting the retrieval. A typical Wikipedia page is composed of four parts: the top of the page, the stringcourse of the left, the body, and the footer. The body contains the

main information of the page. It is also divided into four main parts: the title, the introductory summary, the table of contents and the content itself. The page retrieval targets the body part, mainly the content and the introductory summary. This is done by using the Wikidump function, described in Appendix A

However, we should pay attention during the page retrieval. Some pages do not have a dense introductory summary, while others present relevant information only in the summary and not in the content section. Focusing only on the main content of the requested page may not provide relevant information. It is therefore necessary to combine the introductory summary and the content, to obtain a unique text before starting the summarization process. For instance, the introductory summary section of the page "Eséka"[2] is only one sentence (112 characters with spaces) while the length of the content section of the page is 9762 characters. Another example is the page "Song Bassong"[3] that contains relevant information in the introductory summary section that are not present in the content section. In addition, the total number of characters of the latter page is 345 (below the limit 455), meaning that the content of both sections can be sent back to the user, even without a summarization. Even though such situations are quite rare, the content retrieved from the requested page will go through the summarization process.

3.2.2. Summarization process

- **Model**

The model used for the summarization is built through a transfer learning process on a T5 model. The choice of T5 is justified by the fact it provides a better generalization compared to BERT, as it is demonstrated in [59]. As an encyclopaedia, Wikipedia deals with various topics. Thus, the T5 model is more suitable. In addition, the model can further be used for other aims such as translation and answers to short questions. T5 is based on transformers. A transformer is composed of two types of layers: encoder and decoder layers. A transformer is composed of two types of layers: encoder and decoder layers. An encoder layer aims to encode long sequences input into a numerical form through the attention mechanism, while a decoder layer aims to output a summary by using the encoded information from the encoder layers. Transformers try to solve the parallelization issue by using Convolutional Neural Networks (CNNs) together with attention models. Figure 3 illustrates the data flow in transformer with 6 encoders and 6 decoders. Each encoder has two types of layers: multi-head self-attention layer and position-wise fully connected feed-forward network. In addition to both previous layers, a decoder also includes a masked multi-head self-attention layer [60].

---

[2] https://fr.wikipedia.org/wiki/%C3%89s%C3%A9ka

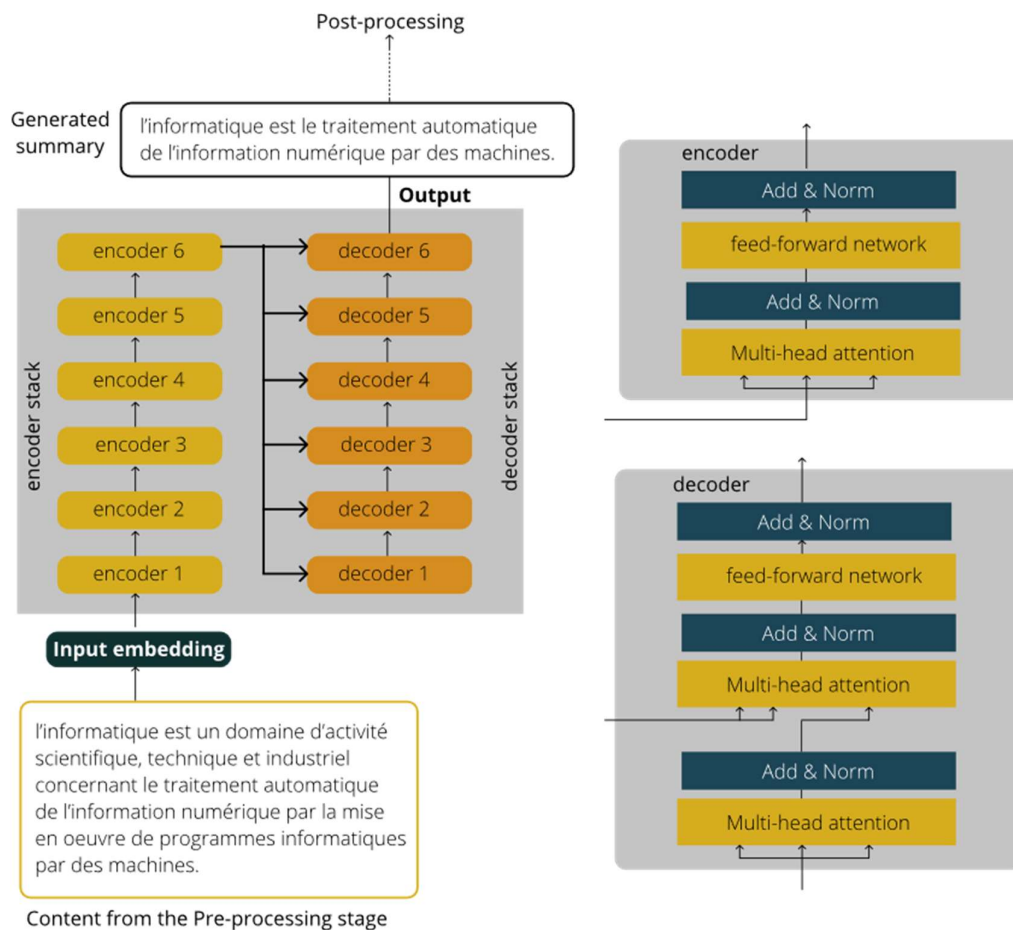[3] https://fr.wikipedia.org/wiki/Song_Bassong

Figure 3. Data flow in transformer.

We make use of the default T5 model with 12 transformer blocks, 220 million parameters, 768 hidden states, 3072 feed-forward hidden-state, 12 self-attention heads. The model has been originally trained on English text using the Colossal Clean Crawled Corpus (C4) introduced in [57]. We retrain the model using a dataset built with Wikipedia pages.

- **Dataset**

To ease the access to data, we choose to create a data frame from text files. two steps are important to generate such a dataset: Understanding the structure of the document and defining the output (the structure of the table). Since the structure of the Wikipedia is known, targeted content can be easily retrieved. Only the introductory summary section and content section are retrieved from the body of the page. This is done using the Wikidump function, which also cleans up the text by removing special characters, images, etc. The structure of the dataset, inspired by the structure of OrangeSum dataset[4], is composed of two main fields: Text (content section) and Summary (introductory summary section). To build the dataset, a crawler starts from a Wikipedia URL and selects 25,000 pages. An excerpt of the dataset is provided in Appendix B1.

- **Training and testing**

We consider the default architecture of the T5 model, and we apply the Adam optimizer proposed in [61] by initializing the learning rate to 0.0001. The input batch size for training and testing is 2, and we use 2 epochs for the training. The training was done using a Tesla T4 from Colab. We split the datasets to 80% for the training set and 20% for the

---

[4] https://huggingface.co/datasets/viewer/?dataset=orange_sum

test set. The Wikipedia summary section was used as a reference summary as it is done in [25].

3.3.3. Post-processing of the summary

The post-processing is done with the help of the *length_Checker* function. It checks the length of the summary and eventually modifies it to get an understandable final summary. In fact, the maximal number of characters (455) is given as a parameter to the model. Then the model outputs the first 455 characters of the generated summary in case the length of the generated summary is greater than 455. After truncation, the last sentence may not be understandable. The easiest way to fix this, is to delete the eventually incomplete last sentence. The *length_Checker* function is described in Appendix C.

4. Results

The well-known ROUGE (Recall-Oriented Understudy for Gisting Evaluation) metrics are used to evaluate the quality of a summary or machine translation [62]. ROUGE works by comparing an automatically produced summary against a set of typically human-produced reference summaries. The measure is done by counting the number of matching words between a generated summary and the reference summary. In other terms, ROUGE measures the content coverage of an automatic summary over the reference summary. Different variants of ROUGE have been proposed such as ROUGE-N, ROUGE-L and ROUGE-S that count respectively the number of overlapping units of n-gram, word sequences, and word pairs between a generated summary and the reference summary [62]. The formula for ROUGE-N is given in equation 1.

$$\text{ROUGE} - \text{N} = \frac{\sum_{gram_n \epsilon S} Count_{match}(gram_n)}{\sum_{gram_n \epsilon S} Count(gram_n)} \, , \tag{1}$$

where gram is the choice of n-gram and S is the reference summary.

4.1. Comparison

4.1.1. Comparison with extractive approaches

We first compare the results of the proposed summarization model with the results in [25] that defined an approach associating the LSA and Text Rank algorithms for the text summarization of Wikipedia articles. However, the tests carried out in [25] are done on English pages. For a fair comparison, we selected the corresponding French Wikipedia pages of articles in [25]. We focused on the results of articles with a size greater than 2,000 characters. This is done for two main reasons. Firstly, most of the French pages in Wikipedia are longer than their English version. For instance, the French version of the page on "Kousseri" is 5,797 characters, while the English version is only 1,386 characters. Secondly, from the results in [25], the proposed extractive approach struggles to provide a good result when the length of the page is increasing. So, the idea is to also appreciate the quality of results provided by the abstractive approach over extractive approaches when dealing with quite long documents. The results of the comparison are provided in Table 1.

Table 1. Comparison of Wikipedia article with length over than 2000 characters.

Titre	Kousseri	Eseka	Tibati	Yaou Aissatou	Bafoussam	Chantal Biya
LA[1] (Fr)	5,797	9,762	2,875	3,000	10,902	4,004
LA (En)	1,386	510	729	1,233	1,790	1,951
RS[2] (Fr)	372	112	831	1560	401	341
RS (En)	459	43	726	1230	1516	100
LT[3] ROUGE-1 (En)	0.57	**1**	0.56	0.68	0.52	**1**
LT ROUGE-2 (En)	0.56	**1**	0.5	0.60	0.45	**1**
WS[4] ROUGE-1 (Fr)	**0.97**	0.76	**0.91**	**0.74**	**0.77**	0.80
WS ROUGE-2 (Fr)	**0.97**	0.75	**0.90**	**0.71**	**0.76**	0.83

From Table 1, we observe that the summary produced by the proposed abstractive model is most of the time better than the one produced by the extractive approach, even though French pages are longer than their corresponding English pages. A deep observation reveals that the extractive approach struggles to provide a good summary when the reference summary is longer. For instance, on the Wikipedia page "Kousseri", the reference summary contains 459 characters and the best ROUGE metric equal to 0.57 for LT ROUGE-1. When the length of the reference summary increases like on the page "Bafoussam", the LT ROUGE-1 drops to 0.52. However, the extractive approach provides better results when the length of the reference summary is very short (up to hundred characters).

Table 2 provides and inspection of some summaries to better appreciate the ability of the proposed abstractive approach to generate new texts. Three cases can be observed. In the first case ("Kousséri"), the summarizer uses partial sentences from both summary and content sections. The generated summary is composed of a first excerpt in the summary section, a second in the content section, and a last in the summary section again. In the second case ("Chantal Biya"), the summary is only composed of partial sentences from the summary section. In the last case ("Informatique"), a new text is inserted in the summary.

Table 2. Excerpt of some Wikipedia pages and their related summaries generated by the proposed approach. Underlined texts in the "Page Content" and "Generated Summary" columns represent excerpts used to generate the summary and new texts inserted by the summarizer respectively.

Title	Page Content	Generated Summary
Kousséri	*Summary section* Kousséri est une commune du Cameroun située dans la région de l'Extrême-Nord et le département du Logone-et-Chari, dont elle est le chef-lieu. À l'époque coloniale, elle se nommait Kusseri au sein de la colonie allemande du Kamerun, *puis Fort-Foureau au sein de la colonie du Cameroun français*. C'est un important marché, notamment pour les échanges avec le Tchad voisin. *Content section* Kousséri est une cité très ancienne, remontant au moins à la civilisation Sao (XIe - XVIe siècles). Ville Kotoko, l'une de cinq principauté "kotoko". Son nom d'origine est M'sr. Un peu plus tard appelé (Qussur : les châteaux) en arabe. Au XVIe siècle, elle était *la capitale d'un royaume de M'sr en langue Kotoko dont les élites au moins étaient islamisées,* et mentionnée sous la forme italienne Uncusciuri par Giovanni Lorenzo d'Anania (en)....	Kousséri est la capitale d'un royaume de M'sr en langue Kotoko dont les élites au moins étaient islamisées, puis Fort-Foureau au sein de la colonie du Cameroun français.
Chantal Biya	*Summary section* Chantal Biya, *née Chantal* Pulchérie *Vigouroux le 4 décembre 1970, est la seconde épouse du second président de la République du Cameroun*, Paul Biya. Elle est la troisième Première dame du Cameroun *depuis son mariage, le 23 avril 1994*. Elle met cette position de Première dame au profit d'un engagement caritatif de lutte contre le SIDA.	née Chantal Vigouroux le 4 décembre 1970, est la seconde épouse du second président de la République du Cameroun depuis son mariage, le 23 avril 1994.
Informatique	*Summary section* *L'informatique est un domaine d'activité scientifique, technique, et industriel concernant le traitement automatique de l'information numérique par* l'exécution *de programmes informatiques par des machines* : des systèmes embarqués, des ordinateurs, des robots, des automates, etc. Ces champs d'application peuvent être séparés en deux branches : théorique : concerne la définition de concepts et modèles ; pratique : s'intéresse aux techniques concrètes de mise en œuvre…	L'informatique est un domaine d'activité scientifique, technique, et industriel concernant le traitement automatique de l'information numérique par *la mise en oeuvre* de programmes informatiques par des machines.

WATS-SMS has also been compared to Wikipedia-based summarizer, another extractive-based summarizer system proposed by Sankarasubramaniam et al [22]. Their

approach consists first to construct a bipartite sentence-concept graph, and to rank the input sentences using iterative updates on this graph. The work in [22] also makes use of English Wikipedia pages. We used ROUGE metrics for the evaluation and computed the recall scores for 100-word summaries to be able to reproduce the test conditions used by Sankarasubramaniam et al [22], with the only difference that our proposed model uses French Wikipedia pages.

Table 3. Comparison with extractive summarizer.

Summarizer	Descriptions	ROUGE-1	ROUGE-2
Wikipedia-based summarizer	Bipartite sentence-concept graph,rank the input sentence	0.46	0.23
WATS-SMS	Models based on Wikipedia Datasets.	0.52	0.41

From Table 3, WATS-SMS provides competitive results, 52% and 41% corresponding respectively to the recall of ROUGE-1 and ROUGE-2. All those results validate the assumption that the abstractive approaches provide better results than extractive approaches.

4.1.2. Comparison with abstractive models

The proposed approach is compared to other abstractive models defined in [63] and trained using English Wikipedia. We first compared our model to seq2seq-attention and Transformer-ED with an input sequence length up to 500 using the ROUGE-L metric, as it is done in the original paper. The results are given in Table 4.

Table 4. Comparison with other abstractive models (length up to 500 characters).

Model	Language	ROUGE-L	L
Seq2seq-attention	English	12.7	500
Transformer-ED	English	34.2	500
WATS-SMS	French	**52.3**	[0-500]

For small documents, the proposed model produces a score of 52.3%, thus showing the efficiency of the model for summarizing French Wikipedia pages.

Finally, the proposed model has been compared with Transformer-DMCA using input text between 10,902 and 11,000.

Table 5. Comparison with other abstractive models (length up to 500 characters).

Model	Language	ROUGE-L	L
Transformer-DMCA, no MoELayer	English	36.2	11,000
Transformer-DMCA, MoE-128	English	37.9	11,000
WATS-SMS	French	**77.0**	10,902

As we observe from Table 5, the proposed model provides a better ROUGE-L value than both Transformer-DMCA without MoE-Layer and Transformer-DMCA with MoE-128. This demonstrates the ability of the proposed model to provide good summary.

5. Demonstration

This section shows a demonstration of WATS-SMS on a phone with limited features. In Figure 4 the user starts by sending a request with the keyword "resume", followed by the title of the requested page: "informatique". Figure 4.b shows the running of the SMS-Gateway application on a smartphone which is used as a gateway, taking the request, and sending to the Summarizer. After receiving the summary, the SMS Gateway sends it back to the user. Figures 4.c shows the summary on the user's device. We notice that the SMS containing the summary is complete since its length is less than 455 characters defined as the maximal length.

(a) Sending a request

(b) Processing the requested page

(c) Displaying the summary

Figure 4. Demo of using WATS-SMS.

6. Conclusions and future works

Despite the enormous progress in the field of automatic NLP, text summarization remains a challenging task. This work has highlighted the text summarization issue of French Wikipedia articles with character limitation. This paper introduced WATS-SMS, an abstractive text summarizer for French Wikipedia pages built through a fine-tuning process using T5 model which is based on the Transformer architecture. After training on 25,000 French pages, the generated model outperforms extractive approaches and provides satisfactory results compared to other abstractive models focusing on the English language. Through the SMS-Gateway, users with basic telephones can benefit from the system.

However, even though the limitation of the number of characters in the summary is given as a parameter to the model, the summary requires a post-processing to avoid an uncomplete summary. In fact, the model most of the time produces summaries with an

incomplete last sentence which is removed during post-processing. It would be interesting to deal with the number of characters limitation directly during the processing stage rather than counting the number of characters in the generated summary and sending a truncated summary. Another important issue is the case of very long pages. Due to the number of characters limitation (455 characters), the summary can miss relevant information. It would be therefore interesting to design guided summarization process that can consider user preferences.

Appendix A: Wikidump function

The Wikidump function uses the wikipedia library (python library) to connect to the wikipedia page and to retrieve the content of the page corresponding to the given title. This content is cleaned up using regular expressions in order to delete unwanted items.

Algorithm A1: Wikidump

Input: *title*: Title of the requested page

Output: *text*: Cleaned Content for summarization process;

Begin

   Import Wikipedia

   wikipedia.set_lang("fr")

   Connect to fr.wikipedia.org/title

   wiki = wikipedia.page(title)

   If a response **then**

      Content = wiki.content

      For every line in the Content

         text = re.sub(r'==.*?==+', '', text)

         text = text.replace('\n', '')

      endFor

   endIf

   return *text*

End

Appendix B: Dataset

Figure B1. An excerpt of the dataset.

Appendix C: Length_Checker function

The Length_checker algorithm aims to ensure that the length of the generated summary is not exceeding the maximal length. When retrieving only 455 characters, we can obtain an incomplete sentence at the end. The main idea is to retrieve a text longer than 455 characters and to delete last sentence until we get a set of sentences which length is not exceeding 455 characters.

Algorithm C1: Length_Checker

Input: $G_{summary}$: The generated summary by the summarizer

Output: $F_{Summary}$: The final summary sent to the user;

Begin

    $F_{Summary} = G_{summary}$

    If length($F_{summary}$) <= 455 **then**

        return $F_{summary}$

    Else

        While $F_{summary}$ > 455 do

            remove_last_sentence($F_{summary}$)

    return $F_{summary}$

End

References

[1] M. T. Maybury, « Generating summaries from event data », *Inf. Process. Manag.*, vol. 31, nᵒ 5, p. 735-751, 1995.

[2] W. S. El-Kassas, C. R. Salama, A. A. Rafea, et H. K. Mohamed, « Automatic text summarization: A comprehensive survey », *Expert Syst. Appl.*, vol. 165, p. 113679, mars 2021, doi: 10.1016/j.eswa.2020.113679.

[3] J. Antunes, R. D. Lins, R. Lima, H. Oliveira, M. Riss, et S. J. Simske, « Automatic cohesive summarization with pronominal anaphora resolution », *Comput. Speech Lang.*, vol. 52, p. 141-164, 2018.

[4] J. Steinberger, M. Kabadjov, M. Poesio, et O. Sanchez-Graillet, « Improving LSA-based summarization with anaphora resolution », in *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, 2005, p. 1-8.

[5]   J. Steinberger, M. Poesio, M. A. Kabadjov, et K. Ježek, « Two uses of anaphora resolution in summarization », *Inf. Process. Manag.*, vol. 43, nᵒ 6, p. 1663-1680, 2007.

[6]   A. Sahni et S. Palwe, « Topic Modeling on Online News Extraction », in *Intelligent Computing and Information and Communication*, Springer, 2018, p. 611-622.

[7]   P. Sethi, S. Sonawane, S. Khanwalker, et R. B. Keskar, « Automatic text summarization of news articles », in *2017 International Conference on Big Data, IoT and Data Science (BID)*, 2017, p. 23-29.

[8]   R. Bhargava et Y. Sharma, « MSATS: Multilingual sentiment analysis via text summarization », in *2017 7th International Conference on Cloud Computing, Data Science & Engineering-Confluence*, 2017, p. 71-76.

[9]   A. J. J. Mary et L. Arockiam, « ASFuL: Aspect based sentiment summarization using fuzzy logic », in *2017 International Conference on Algorithms, Methodology, Models and Applications in Emerging Technologies (ICAMMAET)*, 2017, p. 1-5.

[10] R. Chakraborty, M. Bhavsar, S. K. Dandapat, et J. Chandra, « Tweet summarization of news articles: An objective ordering-based perspective », *IEEE Trans. Comput. Soc. Syst.*, vol. 6, nᵒ 4, p. 761-777, 2019.

[11] G. Carenini, R. T. Ng, et X. Zhou, « Summarizing email conversations with clue words », in *Proceedings of the 16th international conference on World Wide Web*, 2007, p. 91-100.

[12] S. Mohammad *et al.*, « Using citations to generate surveys of scientific paradigms », in *Proceedings of human language technologies: The 2009 annual conference of the North American chapter of the association for computational linguistics*, 2009, p. 584-592.

[13] X.-J. Jiang, X.-L. Mao, B.-S. Feng, X. Wei, B.-B. Bian, et H. Huang, « Hsds: An abstractive model for automatic survey generation », in *International Conference on Database Systems for Advanced Applications*, 2019, p. 70-86.

[14] J.-T. Sun, D. Shen, H.-J. Zeng, Q. Yang, Y. Lu, et Z. Chen, « Web-page summarization using clickthrough data », in *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, 2005, p. 194-201.

[15] J. L. F. K. Ebongue, « Rethinking Network Connectivity in Rural Communities in Cameroon », présenté à IST-Africa, Lilongwe, Malaw, 2015. [En ligne]. Disponible sur: https://arxiv.org/ftp/arxiv/papers/1505/1505.04449.pdf

[16] M. Pinkovskiy et X. Sala-i-Martin, « Africa is on time », *J. Econ. Growth*, vol. 19, nᵒ 3, p. 311-338, 2014.

[17] J. L. F. K. Ebongue, C. Thron, et J. M. Nlong, « Mesh Router Nodes placement in Rural Wireless Mesh Networks », *ArXiv Prepr. ArXiv150503332*, 2015.

[18] J. L. E. K. Fendji, C. Thron, et J. M. Nlong, « A metropolis approach for mesh router nodes placement in rural wireless mesh networks », *ArXiv Prepr. ArXiv150408212*, 2015.

[19] P. Du Plessis et R. Mestry, « Teachers for rural schools–a challenge for South Africa », *South Afr. J. Educ.*, vol. 39, 2019.

[20] A. Mulkeen, « Teachers for rural schools: A challenge for Africa », *Rome FAO*, 2005.

[21] N. Selwyn et S. Gorard, « Students' use of Wikipedia as an academic resource—Patterns of use and perceptions of usefulness », *Internet High. Educ.*, vol. 28, p. 28-34, 2016.

[22] Y. Sankarasubramaniam, K. Ramanathan, et S. Ghosh, « Text summarization using Wikipedia », *Inf. Process. Manag.*, vol. 50, nᵒ 3, p. 443-461, mai 2014, doi: 10.1016/j.ipm.2014.02.001.

[23] K. Ramanathan, Y. Sankarasubramaniam, N. Mathur, et A. Gupta, « Document summarization using Wikipedia », in *Proceedings of the first international conference on intelligent human computer interaction*, 2009, p. 254-260.

[24] C.-Y. Lin et E. Hovy, « The potential and limitations of automatic sentence extraction for summarization », in *Proceedings of the HLT-NAACL 03 Text Summarization Workshop*, 2003, p. 73-80.

[25] J. L. E. K. Fendji et B. A. H. Aminatou, « From web to SMS: A text summarization of Wikipedia pages with character limitation », *EAI Endorsed Trans. Creat. Technol.*, vol. 7, nᵒ 24, Art. nᵒ 24, 2020, doi: 10.4108/eai.11-6-2020.165277.

[26] S. Banerjee et P. Mitra, « WikiWrite: Generating Wikipedia Articles Automatically », p. 7.

[27] R. Paulus, C. Xiong, et R. Socher, « A Deep Reinforced Model for Abstractive Summarization », *ArXiv170504304 Cs*, nov. 2017, Consulté le: juin 21, 2021. [En ligne]. Disponible sur: http://arxiv.org/abs/1705.04304

[28] S. Gehrmann, Y. Deng, et A. M. Rush, « Bottom-Up Abstractive Summarization », *ArXiv180810792 Cs*, oct. 2018, Consulté le: juin 21, 2021. [En ligne]. Disponible sur: http://arxiv.org/abs/1808.10792

[29] F. Liu, J. Flanigan, S. Thomson, N. Sadeh, et N. A. Smith, « Toward Abstractive Summarization Using Semantic Representations », *ArXiv180510399 Cs*, mai 2018, Consulté le: juin 21, 2021. [En ligne]. Disponible sur: http://arxiv.org/abs/1805.10399

[30] Y.-C. Chen et M. Bansal, « Fast Abstractive Summarization with Reinforce-Selected Sentence Rewriting », *ArXiv180511080 Cs*, mai 2018, Consulté le: juin 21, 2021. [En ligne]. Disponible sur: http://arxiv.org/abs/1805.11080

[31] A. Fan, D. Grangier, et M. Auli, « Controllable Abstractive Summarization », *ArXiv171105217 Cs*, mai 2018, Consulté le: juin 21, 2021. [En ligne]. Disponible sur: http://arxiv.org/abs/1711.05217

[32] T. Wolf *et al.*, « Transformers: State-of-the-art natural language processing », in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, 2020, p. 38-45.

[33] H. P. Luhn, « The automatic creation of literature abstracts », *IBM J. Res. Dev.*, vol. 2, nᵒ 2, p. 159-165, 1958.

[34] H. P. Edmundson, « New methods in automatic extracting », *J. ACM JACM*, vol. 16, nᵒ 2, p. 264-285, 1969.

[35] M. Gambhir et V. Gupta, « Recent automatic text summarization techniques: a survey », *Artif. Intell. Rev.*, vol. 47, nᵒ 1, p. 1-66, 2017.

[36] H. Christian, M. P. Agus, et D. Suhartono, « Single document automatic text summarization using term frequency-inverse document frequency (TF-IDF) », *ComTech Comput. Math. Eng. Appl.*, vol. 7, nᵒ 4, p. 285-294, 2016.

[37] K. Sarkar, « Automatic single document text summarization using key concepts in documents », *J. Inf. Process. Syst.*, vol. 9, nᵒ 4, p. 602-620, 2013.

[38] G. Erkan et D. Radev, « Lexpagerank: Prestige in multi-document text summarization », in *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, 2004, p. 365-371.

[39] T. Uçkan et A. Karcı, « Extractive multi-document text summarization based on graph independent sets », *Egypt. Inform. J.*, vol. 21, nᵒ 3, p. 145-157, 2020.

[40] A. Patel, T. Siddiqui, et U. S. Tiwary, « A language independent approach to multilingual text summarization », *Large Scale Semantic Access Content Text Image Video Sound*, p. 123-132, 2007.

[41] D. R. Radev *et al.*, « MEAD-a platform for multidocument multilingual text summarization », 2004.

[42] E. L. Pontes, C.-E. González-Gallardo, J.-M. Torres-Moreno, et S. Huet, « Cross-lingual speech-to-text summarization », in *International Conference on Multimedia and Network Information System*, 2018, p. 385-395.

[43] J. Zhu *et al.*, « NCLS: Neural cross-lingual summarization », *ArXiv Prepr. ArXiv190900156*, 2019.

[44] N. Moratanch et S. Chitrakala, « A survey on extractive text summarization », in *2017 international conference on computer, communication and signal processing (ICCCSP)*, 2017, p. 1-6.

[45] N. Moratanch et S. Chitrakala, « A survey on abstractive text summarization », in *2016 International Conference on Circuit, power and computing technologies (ICCPCT)*, 2016, p. 1-7.

[46] G. Di Fabbrizio, A. Stent, et R. Gaizauskas, « A Hybrid Approach to Multi-document Summarization of Opinions in Reviews », in *Proceedings of the 8th International Natural Language Generation Conference (INLG)*, Philadelphia, Pennsylvania, U.S.A., juin 2014, p. 54-63. doi: 10.3115/v1/W14-4408.

[47] L. Hou, P. Hu, et C. Bei, « Abstractive document summarization via neural model with joint attention », in *National CCF Conference on Natural Language Processing and Chinese Computing*, 2017, p. 329-338.

[48] S. Wang, X. Zhao, B. Li, B. Ge, et D. Tang, « Integrating extractive and abstractive models for long text summarization », in *2017 IEEE International Congress on Big Data (BigData Congress)*, 2017, p. 305-312.

[49] P. Liu, X. Qiu, X. Chen, S. Wu, et X.-J. Huang, « Multi-timescale long short-term memory neural network for modelling sentences and documents », in *Proceedings of the 2015 conference on empirical methods in natural language processing*, 2015, p. 2326-2335.

[50] S. Hochreiter et J. Schmidhuber, « Long short-term memory », *Neural Comput.*, vol. 9, nᵒ 8, p. 1735-1780, 1997.

[51] A. Vaswani *et al.*, « Attention is all you need », in *Advances in neural information processing systems*, 2017, p. 5998-6008.

[52] J. Devlin, M.-W. Chang, K. Lee, et K. Toutanova, « Bert: Pre-training of deep bidirectional transformers for language understanding », *ArXiv Prepr. ArXiv181004805*, 2018.

[53] L. Martin *et al.*, « Camembert: a tasty french language model », *ArXiv Prepr. ArXiv191103894*, 2019.

[54] H. Le *et al.*, « Flaubert: Unsupervised language model pre-training for french », *ArXiv Prepr. ArXiv191205372*, 2019.

[55] Y. Liu et M. Lapata, « Text summarization with pretrained encoders », *ArXiv Prepr. ArXiv190808345*, 2019.

[56] Y. Liu, « Fine-tune BERT for extractive summarization », *ArXiv Prepr. ArXiv190310318*, 2019.

[57] C. Raffel *et al.*, « Exploring the limits of transfer learning with a unified text-to-text transformer », *ArXiv Prepr. ArXiv191010683*, 2019.

[58] « Google AI Blog: Exploring Transfer Learning with T5: the Text-To-Text Transfer Transformer ». https://ai.googleblog.com/2020/02/exploring-transfer-learning-with-t5.html (consulté le août 26, 2021).

[59] M. Kale, « Text-to-text pre-training for data-to-text tasks », *ArXiv Prepr. ArXiv200510433*, 2020.

[60] « Affine - Bidirectional Encoder Representations for Transformers (BERT) Simplified », *Affine*, déc. 18, 2019. https://www.affine.ai/bidirectional-encoder-representations-for-transformers-bert-simplified/ (consulté le août 26, 2021).

[61] D. P. Kingma et J. Ba, « Adam: A method for stochastic optimization », *ArXiv Prepr. ArXiv14126980*, 2014.

[62] C.-Y. Lin, « Rouge: A package for automatic evaluation of summaries », in *Text summarization branches out*, 2004, p. 74-81.

[63] P. J. Liu *et al.*, « Generating wikipedia by summarizing long sequences », *ArXiv Prepr. ArXiv180110198*, 2018.