# A comparison between Abstractive and Semi-Abstractive methods for text summarization on the CNN/Daily Mail dataset

MSc Thesis

**Ioannis Tsogias**
6966985
i.tsogias@students.uu.nl

**First Supervisor:** Dr. M.P. (Marijn) Schraagen
**Second Supervisor:** Dr. S. (Shihan) Wang

Artificial Intelligence, Utrecht University
The Netherlands
July 12, 2021

# Acknowledgements

I would like to express my very great appreciation and gratitude to my research supervisor, for his patient guidance and constructive feedback of this paper and keeping my progress on schedule. My grateful thanks are extended to all 10 participants of the experiment that was conducted for the needs of this research, for their time and patience which were needed to complete the whole survey. I would also like to thank my family, for their continuous support of my work and keeping me motivated. Lastly, my friends were of great support in deliberating over my problems and findings, as well as providing happy distraction to rest my mind outside of my research.

# Abstract

In this day and age, there is an enormous amount of textual material, and it is only growing every single day. There is a great need to reduce much of this text data to shorter, focused summaries that capture the salient details.

The main goal of this research is to examine what type of summaries do human users prefer. Thus, we compare the performance of a straightforward Attention-Based Encoder-Decoder abstractive summarization method, against a hybrid model that firstly detects the most important parts of the original document, and then uses paraphrasing in order to create a novel summary, on the non-anonymized version of the CNN/Daily dataset. Besides ROUGE, that is being used as a baseline automatic evaluation metric, we conduct a human evaluation experiment in order to assess the Readability and Relevance of the output summaries of our two main models, three extractive baseline models and reference summaries. The level of correlation of ROUGE and Human Evaluation results is being measured alongside with the level of abstractiveness of the models' outputs.

Both ROUGE and Human Evaluation results indicate that the abstractive approach outperforms the hybrid method. However, the correlation analysis showcases that the automatic and human evaluation results have no relationship. Furthermore, our abstractiveness results showcase that both models tend not to use novel words, but rather copy from the original text, which can be explained by the low abstractiveness scores of the reference summaries on which our models are trained, a fact that makes the appropriateness of the CNN/Daily Mail dataset for the task highly questionable.

Through this study, we argue that despite that ROUGE can be useful to quickly assess the quality of a summary, it mainly provides information regarding its informativeness. However, there are more things to consider when evaluating a summary such as text's fluency, grammaticality, factual accuracy, completeness, length and abstractiveness.

# Contents

# 1 Introduction

Scientists have been studying the human language since the early 1900s when a Swiss linguistics professor named Ferdinand de Saussure died, and in the process, almost deprived the world of the concept of Language as a Science. Two of his colleagues, Albert Sechehaye and Charles Bally, recognized the importance of his concepts and took the unusual steps of collecting his notes for a manuscript and his students' notes from the courses Saussure had been teaching. From these, they wrote "Cours de Linguistique Générale" (de Saussure, 1916). The book laid the foundation for what has come to be called the structuralist approach, starting with linguistics, and later expanding to other fields, including Computer Science.

Later, in 1950 Alan Turing introduced the concept of a "thinking" machine (Turing, 1950). His idea proposed that if a machine could be part of a conversation through the use of a teleprinter, and it has the ability to imitate a human completely, so there were no noticeable differences, then the machine could be considered capable of thinking. Shortly after this, in 1952, the Hodgkin-Huxley model showed how the brain uses neurons in forming an electrical network. These events helped inspire the idea of Artificial Intelligence (AI), Natural Language Processing (NLP), and the evolution of Computer Science as a whole.

**NLP** is a subfield of **Linguistics**, **Computer Science**, and **Artificial Intelligence** concerned with the interactions between computers and human language that helps computers understand, interpret, and utilize human languages. NLP allows computers to communicate with people, using a human language and provides computers with the ability to read text, hear speech, and interpret it. NLP draws from several disciplines, including computational linguistics and computer science, as it attempts to close the gap between human and computer communications. In general terms, NLP breaks down language into shorter, more basic pieces, called tokens (words and non-word characters such as periods, word n-grams, character n-grams, sentences, etc.), and attempts to understand the relationships between those tokens. These are being used by higher-level NLP processes such as Machine Translation, Sentiment Analysis, Topic Modelling and many more.

In this day and age, there is an enormous amount of textual material, and it is only growing every single day. The internet, comprised of news articles, blogs, and many other types of online content is the main source of unstructured textual data. Thus, there is a great need to reduce much of this text data to shorter, focused summaries that capture the salient details, both so we can navigate it more effectively as well as check whether the larger documents contain the information that we are looking for. As the reader may understand, the outcomes of research related to Automatic Text Summarization can eventually become quite important for people as in a future

scenario where automatic text summarizers will be able to generate reliable, fluent and accurate summaries can facilitate people's lives as it can save everyone time. The author(s) of some text can use Artificial Intelligence products regarding Text Summarization in order to provide the readers with summaries, while the readers will be able to get the salient parts of the original text without having to read the whole text.

In more detail, **Text Summarization** is the process of distilling the most important information from a source to produce an abridged version for a particular user and task (Mani and Maybury, 1999). Generally, there are two main approaches to summarize text documents. The first one is the **Extractive** method which involves the selection of phrases and sentences from the source document to create the new summary. On the other hand, **Abstractive** summarization involves generating entirely new phrases and sentences to capture the meaning of the source document. This is a more challenging approach, but is also the approach used by humans more often in real life.

Both of the above mentioned methods have their advantages and disadvantages. While extractive summarizers produce more readable summaries as they use already written sentences that follow grammatical and syntactical rules, abstractive summarizers can present the information from the source text in a compact and coherent way due to their ability not to bound by the original sentences. They can shorten sentences, rearrange and combine information and use synonyms while their methodology correlates more with the way that actual humans summarize texts as in highlighting the most important aspects of the source document, and then try to compress this information into a smaller text by paraphrasing, removing or adding new words or phrases. However, sometimes automatic abstractive summarizers can make mistakes or repeat certain words/ phrases that may lower the level of naturality of the generated text.

This research will focus on the comparison of a straightforward Attention-Based Encoder-Decoder abstractive summarization method, against a hybrid model that firstly detects the most important parts of the original document, and then uses paraphrasing in order to create a novel summary. These two models will be tested, evaluated and compared with each other, as well as with several baselines (e.g. LEAD-3 benchmark (Nallapati et al., 2017)) in the CNN/Daily Mail dataset, using the *ROUGE* metric. *ROUGE* will work as a baseline metric, when comparing the generated summaries with the reference ones, since several researchers have pointed out that it is not a good summarization evaluation metric (Huang et al., 2020; Schluter, 2017). Thus, a human experiment will be conducted that will focus on asking human participants to read and evaluate the generated summaries according to their readability and relevance. This way we will be able to find out which models perform better according to human participants and examine whether human evaluation correlates positively or negatively with the *ROUGE* metric results. Intuitively, the fact that an automatically generated summary's n-grams overlap with some n-grams of the reference summary does not mean that this summary is actually good. This is the main reason why the correlation analysis will be performed. Knowing what kind of summaries users prefer can help, by suggesting preferred methods to be investigated in further research.

This MSc Thesis aims to answer the following research questions:

- How do extractive, abstractive and hybrid summarization models perform on the CNN/Daily Mail dataset?
  - What are the *ROUGE* scores of the extractive baselines?
  - What are the *ROUGE* scores results of the Abstractive Encoder-Decoder Attention based model?
  - What are the *ROUGE* scores results of the Hybrid model?
  - What kind of summaries do human subjects prefer?
- How should automatically generated summaries be evaluated?
  - Is *ROUGE* metric suitable for this task?
  - Does *ROUGE* results correlate with human evaluation results?
  - Is human evaluation a better way to evaluate such systems?

To achieve that, we conduct a scientific survey of the literature in the field of study that is presented in Chapter 2, which discusses the preliminary literature on text summarization as well as the most popular datasets that are being used in previous research. Chapter 3 introduces the scientific method that will be followed in order to collect all results efficiently. Chapter 4 gives an overview of the automatic and human evaluation results while Chapter 5 discusses these results and implications of this research.

# 2 Literature Study

This Chapter discusses the related work around automatic text summarization. Firstly, the most popular datasets that are used for text summarization will be investigated as well as the current summaries' evaluation methods. Then, extractive, abstractive and hybrid summarization methods will be presented alongside with the state of the art models.

## 2.1 Datasets

Data is the backbone of Artificial Intelligence and Machine Learning as without it, it would not be possible to train a machine that learns from and predicts for humans. Textual data collected from single or multiple sources are usually available in unstructured format, which is not practical for machines. But when such data is labeled or tagged with annotation it becomes useful for training a model.

Text summarization is a difficult task as the machine is being called to generate novel summaries given the source document. It's the nature of the human language that makes it difficult. The rules that dictate the passing of information using natural languages are not easy for computers to understand. Thus, good performance requires a proper dataset as it is the case for all ML tasks. This Section presents some of the existing datasets used for text summarization in previous research.

To begin with, the most popular dataset for this task is the **CNN/Daily Mail dataset** as processed by Nallapati et al. (2016a). It contains online news articles (781 tokens on average) paired with multi-sentence summaries (3.75 sentences or 56 tokens on average, 7% of the original text). There are two different versions of this dataset. The first one, introduced by See et al. (2017), uses actual entity names (non-anonymized) while the second one replaces entity occurences with document-specific integer-IDs beginning from 0 (anonymized version, AV). The processed version contains 287,226 training pairs, 13,368 validation pairs and 11,490 test pairs.

Another text summarization dataset is **Gigaword** which has been first used by Rush et al. (2015) and represents a sentence summarization/headline generation task with very short input documents (31.4 tokens) and summaries (8.3 tokens). It contains 3.8M training, 189k development and 1951 test instances.

Similar to Gigaword, task 1 of **DUC 2004** (`https://duc.nist.gov/data.html`) is a sentence summarization task. The dataset used newswire/paper documents

from the TDT and TREC collections and contains 500 documents with average 35.6 tokens and summaries with 10.4 tokens. Due to its size, neural models are typically trained on other datasets and only tested on DUC 2004 (and **DUC 2003**).

**Newsroom** (Grusky et al., 2018) is a summarization dataset of 1.3 million articles and summaries written by authors and editors in newsrooms of 38 major news publications. Extracted from search and social media metadata between 1998 and 2017, these high-quality summaries demonstrate high diversity of summarization styles. In particular, the summaries combine abstractive and extractive strategies, borrowing words and phrases from articles at varying rates. Newsroom contains 1,321,995 articles (Vocabulary size = 6,925,712 words) with the 995,041 of them to be the training set. The mean article length is 658.6 words and the mean summary length is 26.7 words.

Table 2.1 below, presents an overview of the most popular summarization data sets.

| Dataset | Description | Size |
|---|---|---|
| CNN/Daily Mail | News articles | 287K |
| Gigaword | News articles | 3.8M |
| Newsroom | News articles | 1.3M |
| DUC 2003 & 2004 | News articles | 500 |

Table 2.1: An overview of the most popular summarization data sets.

### 2.1.1 Datasets' appropriateness for the task

Nevertheless, not all datasets are appropriate for text summarization. As Bommasani and Cardie (2020) mention in their recent work, high quality data forms the bedrock for building meaningful statistical models in NLP. Thus, data quality must be evaluated either during dataset construction or post hoc. Furthermore, this work suggests that nuanced understanding of data is requisite for drawing sound scientific conclusions and that the correctness and quality of data inherently bounds what can be learned from the data about the task of interest. To explain this, they use an information-theoretic perspective as follows in Figure 2.1 and hypothesize that the quality of the training dataset $T$ is highly correlated with its mutual information with respect to the summarization task $I$, $I(S;T)$, relating this way the data and model quality.

$$\underbrace{I(S;M)}_{\text{learned model}} \leq \underbrace{I(S;T)}_{\text{training data}} + \underbrace{I(S;P)}_{\text{pretraining}} + \underbrace{I(S;A)}_{\text{inductive bias}}$$

Figure 2.1: $I$ denotes the mutual information, $S$ denotes understanding of the underlying summarization task and $M$ denotes a model learned using summarization training data $T$, additional pretraining data $P$, and the model's architecture $A$.

However, data quality has gone largely unquestioned for many recent summarization datasets. In this work, the researchers study compression, topic similarity, abstractivity, redudancy and semantic coherence in many datasets. The research focuses

on these properties, that were chosen as the best among many metrics. Also, for each abstract property, numerous concrete methods can be proposed to quantify it, but the discussion is being restricted to the best performing approaches. The intersection of the datasets studied here and by Bommasani and Cardie contain the CNN/Daily Mail dataset and Gigaword. Their results indicate that Gigaword has the lowest compression scores and that it should not be seen as a summarization dataset but as headline generation dataset that is more in the style of sentence compression. As for the CNN/Daily Mail dataset, researchers concluded that it is suboptimal for studying abstractive systems and that it is not a representative benchmark for summarization as a whole. Despite these issues with the dataset, it is still widely used. The current research will use it to allow comparison with earlier work and easy computation.

## 2.2 Evaluation methods

This Section will showcase an overview of the metrics used to evaluate automatically generated summaries. An exhaustive literature review on summaries' evaluation is beyond the scope of this thesis, but this Section will consider a wide range of studies, focusing on the most popular and recent evaluation metrics used, alongside with their limitations.

Automatically-generated summaries can be evaluated according to intrinsic criteria, which directly relate to the quality of summarization, or extrinsic criteria, that are concerned with the function of the system in which the summaries are used (Walter, 1998). Intrinsic measures, such as *ROUGE*, evaluate the performance of the summarization system in terms of how well the information content of an automatic summary matches the information content of single or multiple human-written summaries. Most of such intrinsic measures possess the advantages of being easy to reproduce and automatic to run. On the other hand, extrinsic methods usually require human subjects to perform a task using different forms of summaries. Extrinsic evaluations are more expensive, since in addition to the human effort required to perform the extrinsic task, the evaluation of task performance is often subjective (Murray et al., 2009). There are cases that extrinsic evaluations are performed automatically, but most research conducts it using human subjects.

### 2.2.1 Automatic Evaluation

To begin with, in summary content evaluation, the summaries are being assessed based on how much relevant information they contain from the source documents. Theoretically, a summary can contain all relevant information from the original document given it is long enough, as it may be almost equal to the original text (Lloret et al., 2018). But the point of summarization is to compress the original text, so an automatic summary must be given a length constraint. Thus, short length and most important and relevant information are the two main requirements that have to be fulfilled. Related work has investigated various metrics to determine whether automatic summaries fulfill this requirement. Earlier studies used metrics

adapted from information retrieval(IR) such as recall, precision and F-measure to assess the content of automatic summaries (e.g. Donaway et al. (2000)). With these metrics, an automatic summary is compared to a human-written one (reference model), and the common sentences between them are measured. The main issue with this kind of metrics is that comparing the generated summary with a single human-written one can be subjective since there can be other sentences within the source document which may be of the same relevance as the ones included in the single model summary. But, since such sentences are not included in the model summary, peer summaries containing such sentences will be scored low although they are as good as the automatic summaries containing sentences from the model summary.

To surpass this issue, McKeown et al. (1998) proposed to use multiple human-written summaries generated by different human subjects and construct an "ideal" summary from these multiple model summaries where the ideal summary is constructed by taking the majority opinion from the multiple summaries. Later, Radev and Tam (2003) proposed relative utility which is a method that multiple judges ranked each sentence of the source document with a score (0-10) that corresponded to its suitability for a summary. Thus, summaries containing different sentences with the same relative utility weights are considered equally good. Only summaries containing sentences with higher relative utility scores are better or scored higher than summaries with less higher relative utility weights.

### 2.2.2 ROUGE (Recall-Oriented Understudy for Gisting Evaluation)

However, both writing multiple summaries for each document and ranking each sentence can be very tedious and time consuming. To tackle this, there was a shift from full sentence comparison to comparison of smaller units inside each sentence. For determining the quality of automated summaries based on n-grams, Lin (2004b), who was inspired by **BLEU** (Papineni et al., 2002) which is a method for automatically evaluating the output of a machine translation system, introduced the most popular summary evaluation system called **ROUGE** which stands for *Recall-Oriented Understudy for Gisting Evaluation*. It is essentially a set of metrics for evaluating automatic summarization of texts as well as machine translation, it is based on precision and recall and works by comparing the similarity of an automatically produced summary or translation with a set of reference summaries (typically human produced).

The $ROUGE_N$ recall score measures the percentage of words in the reference summary that are also present in the generated summary. Here, N is usually equal to 1 or 2 depending on whether we want to work with unigrams ($ROUGE_1$) or bigrams ($ROUGE_2$). Besides $ROUGE_N$, there exist more $ROUGE$ metrics:

- $ROUGE_L$: Measures the Longest Common Subsequence.
- $ROUGE_W$: Measures the overlap of longest common subsequences, favors consecutive matches.

- $ROUGE_S$: Skip-bigram co-occurrence statistics that measure the overlap of skip bigrams between the candidate summary and a set of reference summaries.
- $ROUGE_{SU}$: Obtained from $ROUGE_S$, by adding a begin-of-sentence marker at the beginning of candidate and reference sentences.

---

**ROUGE example**

Here we present an example of the way that $F_1$ $ROUGE_2$ works. Let's assume that we have the following system and reference summaries:

- System Summary: "the cat was found under the bed"
- Reference Summary: "the cat was under the bed"

The bigrams of these sentences are:

- System Summary Bigrams: "the cat", "cat was", "was found", "found under", "under the", "the bed"
- Reference Summary Bigrams: "the cat", "cat was", "was under", "under the", "the bed"

What is needed at this point is to calculate the recall and precision scores. Recall equals to the number of overlapping bigrams divided with the total bigram count of the reference summary, while precision equals to the number of overlapping bigrams divided with the total bigram count of the system summary. In this case, recall = 4/5 and precision = 4/6. Thus, we can calculate $F_1$ which equals to 8/11 (which is approximately equal to 0.72)

---

Despite that $ROUGE$ is the most commonly used evaluation metric for automatically generated summaries, its suitability for the task has been widely criticized. According to Schluter (2017), "Overall quality assurance is problematic: there is no upper bound on the quality of summarization systems, and even humans are excluded from performing optimal summarization". Furthermore, she points out that since $ROUGE_N$ returns the average of scores over multiple reference summaries in order to avoid bias, it is impossible to achieve 100% score. The only situation this can happen is the odd case where all reference summaries contain the exact same n-grams. Thus, a perfect score does not exist, which makes it hard for scientists to know whether there is room for improvement or not.

The main difficulty of the approaches based on n-gram comparison is that they do not account for the fact that relevant information can be expressed using different words or phrases, which is a crucial issue of $ROUGE$ (and other similar metrics) that mainly applies to abstractive summarization (Murray et al., 2009). To tackle this issue, Nenkova and Passonneau (2004) propose the Pyramid method, which uses

variable-length substantial units for comparing generated summaries with human model summaries. These semantic content units are being derived from human annotators who analyze all the model summaries for units of meaning, with each semantic content unit being weighted by how many model summaries it occurs in. Despite Pyramid solving the above mentioned issue, it requires a great deal of human annotation, which is its main drawback.

However, *ROUGE* is a metric used to evaluate the performance of the summarization system based on how well the information represented in the generated summary matches the content of the gold standard summary. Most of such intrinsic measures are invaluable for development purposes, and possess the advantages of being easy to reproduce and automatic to run, according to Murray et al. (2009), which makes it the standard metric for summaries' evaluation.

### 2.2.3 Human evaluation

It is worth noting that, although *ROUGE* is still the most common tool used for content evaluation, this doesn't mean that it is the only or best metric (Lloret et al., 2018). Several metrics have been suggested by the scientific community (e.g. AutoSummENG, MeMoG, FRESA or the Trivergent model), but the main criterion for whether a metric is good or not, stems from the level of correlation with human evaluation, which makes human evaluation the safest way to evaluate machine generated summaries.

According to van der Lee et al. (2019), currently, there is little agreement as to how Natural Language Generation (NLG) systems should be evaluated, with a particularly high degree of variation in the way that human evaluation is carried out. Even though automatic text generation has a long tradition, going back at least to Peter (1677) , human evaluation is still an understudied aspect. However, such an aspect is crucial since with a well-executed evaluation it is possible to assess the quality of a system and its properties, and to demonstrate the progress that has been made on a task, but it can also help us to get a better understanding of the current state of the field.

Human evaluation of natural language generation systems can be done using intrinsic and extrinsic methods. Intrinsic approaches aim to evaluate properties of the system's output, for instance, by asking participants about the fluency of the system's output in a questionnaire, while extrinsic approaches aim to evaluate the impact of the system, by investigating to what degree the system achieves the overarching task for which it was developed. While extrinsic evaluation has been argued to be more useful (Reiter and Belz, 2009), it is also rare. Most research uses intrinsic evaluation since extrinsic evaluation is the most time- and cost-intensive out of all possible evaluations.

Some researchers use expert human subjects on a certain field for human evaluation while others use people that satisfy certain criteria. With an expert-focused design, a small number of expert annotators is recruited to judge aspects of the NLG system. A reader-focused design entails a typically larger sample of (non-expert) participants. Lentz and De Jong (1997) found that these two methods can

be complementary: expert problem detection may highlight textual problems that are missed by general readers. However, this strength is mostly applicable when a more qualitative analysis is used, whereas most expert-focused evaluations in our sample of papers used closed-ended questions with Likert scales.

Evidence suggests that expert readers approach evaluation differently from general readers, injecting their own opinions and biases (Amidei et al., 2018). This might be troublesome if a system is meant for the general population, as expert opinions and biases might not be representative for those of non-experts. This is corroborated by Amidei et al. (2018), who found that expert judgments only predict the outcomes of reader focused evaluation to a limited extent. Experts are also susceptible to considerable variance, so that automatic metrics are sometimes more reliable (Belz and Reiter, 2006). Thus, the conclusion of Belz and Reiter (2006) in favour of large-scale reader-focused studies, rather than expert-focused ones, seems well-taken.

Moving on to certain recent studies that conducted human evaluation, Chen and Bansal (2018) conducted human evaluation to ensure robustness and readability of the generated summaries. In this work, they measured **Relevance** and **Readability**, where the first term is based on the summary containing salient information from the input article, being correct by avoiding contradictory or unrelated information, and avoiding repeated/redundant information. The later one is based on the summary's fluency, grammaticality and coherence. To evaluate both of these criteria, they designed an Amazon MTurk experiment by randomly selecting 100 samples from the CNN/Daily Mail test set and asked human testers to rank between summaries that were generated from different models.

In a very similar manner, Fan et al. (2018), conducted a human evaluation study using Amazon MTurk. Here, they used 500 articles that were randomly selected from the test set of CNN/Daily Mail corpus and showed them to 5 human readers who were given the first 400 words of each news article and were asked to select the summarization output they preferred. In both cases, evaluation showed that *ROUGE* scores agreed with the human evaluation results and the models created at both of these works outperformed the models that they were compared with. Therefore, their models were proven to be able to improve summary quality in a discernible way, compared with previous work.

In another recent work by Liu and Lapata (2019), authors evaluated system output by eliciting human judgements in addition to automatic evaluation. Here, the first task of the evaluation included a question-answering (QA) paradigm which quantifies the degree to which summarization models retain key information from the document. It involved human subjects answering questions, that were created based on the gold summary under the assumption that it highlights the most important document content. Participants, were asked to answer these questions by reading the system summaries, without access to the article. The more questions a system summary can answer, the better it is at summarizing the document as a whole. In this work, they also assessed the overall quality of the summaries by presenting to the participants the output of two systems as well as the original document and let them rank the summaries according to the criteria of Informativeness, Fluency, and Succinctness. Both types of evaluation were conducted on the Amazon

MTurk platform. As shown in their results, participants overwhelmingly prefer the output of BERTSUM against comparison systems across datasets and evaluation paradigms.

## 2.3 Extractive summarization

The **Extractive summarization** technique consists of selecting the most important keywords, sentences, paragraphs etc, from the original document and concatenating them into a shorter form. This Section discusses some of the most popular techniques, presented in previous literature, for sentence extraction.

### 2.3.1 Graph based approach

Graphs have proven to be very helpful in efficiently representing document structure. The general idea is the construction of graphs to capture the relationships between sentences. Early work on graph based extractive summarization (Page et al., 1999) suggested PageRank, which is a method for rating Web pages, measuring human interest and attention devoted to them. A later study created TextRank (Mihalcea and Tarau, 2004), which was essentially based on PageRank, and consists of a graph based ranking model for graphs extracted from natural language text. This algorithm ranks the sentences that are similar to many others as very important. In more detail, in order to find the most relevant sentences in text, a graph is constructed where the vertices of the graph represent each sentence in a document and the edges between sentences are based on content overlap, namely by calculating the number of words that 2 sentences have in common. Based on this network, the sentences are fed into the Pagerank algorithm which identifies the most important sentences. Thus, when one wants to extract a summary of the text, he/she can only take the most important sentences. This algorithm is also widely used for keyword extraction, but our main focus will be on its sentence extraction capabilities.

Later, LexRank was introduced by Erkan and Radev (2011), where the salience of each sentence is determined by the concept of eigenvector centrality. All sentences in each document are represented as a graph and the edges between the sentences represent weighted cosine similarity values. The algorithm creates sentence similarity-based clusters and then ranks the sentences based on their LexRank score, similar to PageRank, except that the similarity graph is undirected in LexRank.

### 2.3.2 Fuzzy logic based approach

Fuzzy logic is a form of logic in which the truth values of variables may be any real number between 0 and 1. It is employed to handle the concept of partial truth, where the truth value may range between completely true and completely false. This approach contains mainly the following four components: defuzzifier, fuzzifier, fuzzy knowledge base and inference engine (Moratanch and Chitrakala, 2017). The textual features that are being used as input in this approach are sentence length, sentence similarity etc (Suanmali et al., 2009a). The initial step for this approach was made

by Suanmali et al. (2009b). In this work, the text document was pre-processed and title features, sentence length, sentence position, term weight, sentences' similarity etc are being extracted. The generated summary follows the sentence ordering of the original document in order to maintain coherency. This method showed improvement in summaries' quality but issues like dangling anaphora, which is the existence of expressions whose antecedent are not being included in the summary, were not addressed.

### 2.3.3 Concept based approach

In the Concept Based Approach information is being extracted from external knowledge bases like HowNet or Wikipedia. The first step in this approach involves retrieving concepts of a text from external knowledge bases (HowNet, WordNet, Wikipedia) and then building a conceptual vector or graph model to depict relationships between concepts and sentences. Then, a ranking algorithm is being applied to score sentences following the generation of summaries based on the ranking scores of sentences (Moratanch and Chitrakala, 2017). Wang et al. (2005) proposed a methodology where the importance of sentences is calculated based on the concepts retrieved from HowNet, instead of words. A conceptual vector model is built to obtain a rough summarization and similarity measures are calculated between the sentences to reduce redundancy in the final summary. A similar work by Sankarasubramaniam et al. (2014) proposed a Wikipedia-based summarization which utilizes graph structure to produce summaries. This method uses Wikipedia to obtain concepts for each sentence and builds a sentence-concept bilateral graph.

### 2.3.4 Semantic text matching

Semantic text matching is a technique used in NLP to identify text information which is semantically related. Zhong et al. (2020), instead of following the commonly used framework of extracting sentences individually and modeling the relationship between sentences, proposed the formulation of the extractive summarization task as a semantic text matching problem, in which a source document and candidate summaries (extracted from the original text) are being matched in a semantic space (Figure 2.2).

Their model is named **MatchSum** and the principle idea here, is that a good summary should be more semantically similar as a whole to the source document than the unqualified summaries. To achieve that, they propose a Siamese-BERT architecture to compute the similarity between the source document and the candidate summary. Siamese BERT leverages the pre-trained BERT (Devlin et al., 2018) in a Siamese network structure (Bromley et al., 2001; Hoffer and Ailon, 2014; Reimers and Gurevych, 2019) to derive semantically meaningful text embeddings that can be compared using cosine-similarity. The Siamese-BERT, used in this study, consists of two BERTs with tied-weights and a cosine-similarity layer during the inference phase.

The matching idea, while intuitive, suffers from combinatorial explosion problems. For example, how could the size of the candidate summary set be determined or

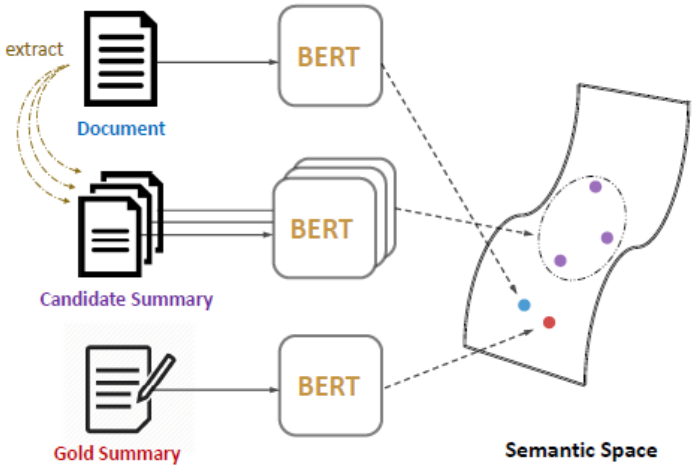


Figure 2.2: MatchSum framework. The contextual representations of the original document are being matched with the gold summary and the candidate summaries (extracted from the document). Intuitively, better candidate summaries should be semantically closer to the document, while the gold summary should be the closest.

should all possible candidates be given a score? To alleviate these difficulties, Zhong et al. (2020) propose a simple candidate pruning strategy where a content selection module is being introduced, to pre-select salient sentences. The module learns to assign each sentence a salience score and prunes sentences irrelevant with the current document, resulting in a pruned document. This content selection module is a parameterized neural network, BERTSUM (Liu and Lapata, 2019), without trigram blocking (Trigrams are blocked during the beam search (Paulus et al., 2017)) in order to score each sentence. Then, a simple rule to obtain the candidates is being used: generating all combinations of selected sentences subject to the pruned document, and reorganizing the order of sentences according to the original position in the document to form candidate summaries. Experimental results show MatchSum outperforms the current state-of-the-art extractive model on six benchmark datasets (in which CNN/Daily Mail is included), which demonstrates the effectiveness of this method (Table 2.2).

### 2.3.5 Sentence ranking

Sentence ranking is a method that includes ranking each sentence of the input document by assigning weights. Highly ranked sentences are extracted from the input document and are added to the output summary. Early work on this approach (Salton and Buckley, 1988) suggested that high word frequency in the input document and low overall word frequency in the whole corpus shows the importance of each term. Later, Goldstein and Carbonell (1998), proposed a novel method for a search engine, which was applied to summarization, that takes into account both relevance to the query and the existence of similarity to the previously selected document, at least at a minimal level. Nomoto and Matsumoto (2001), introduced a K-means clustering algorithm to detect diverse topic windows in the text and used

TFIDF (Jones, 1972) to identify the most important sentences in each cluster.

Ferreira et al. (2013) performed a quantitative and qualitative evaluation of algorithms that were suggested up to this year for sentence summarization that included word scoring, sentence scoring and graph scoring. Their results in three different datasets that included news articles, scientific papers and blogs, showcased that word frequency, TFIDF, sentence position and sentence length were the top performers. However, performance was highly dependent on the dataset that was used.

The evolution of neural networks led to them being applied in text summarization. Cao et al. (2015) introduced a framework for multi-document summarization, using a RNN to rank sentences according to a list of 23 word and sentence level features. A little later, Nallapati et al. (2017), introduced SummaRuNNer, a Recurrent Neural Network (RNN) based sequence-to-sequence model for extractive summarization. A novel contribution of this work was the abstractive training of the extractive model which can train on human generated reference summaries alone, eliminating the need for sentence-level extractive labels. Nallapati et al. (2016b) introduced the framework "Classify or Select" motivated by two intuitive strategies humans tend to adopt when being called to extract salient sentences from a text document. The first strategy, Classify, involves parsing the whole document and then traversing through the sentences in order to decide whether each one of them belongs to the summary or not, while Select involves memorizing the whole document and then picking sentences that should be added to the summary one at a time.

Cao et al. (2016) proposed AttSum, an attention based model that simulates reading habits of humans. AttSum aims to create reference summaries that perform well on query relevance ranking and sentence saliency. Cheng and Lapata (2016) presented a word-based model that operates over continuous representations, produces multi-sentence output, and jointly selects summary words and organizes them into sentences. For this to be done, researchers proposed a hierarchical document encoder and an attention-based extractor. In the same spirit, Zhou et al. (2018) proposed NeuSum, a NN model that learns to jointly score and select sentences which was considered as a novel work since previous research treated these two as individual tasks.

### 2.3.6 State of the art

Table 2.2 below presents the most important methods from related work, including *ROUGE* scores for each one of them.

| Method | ROUGE-1 | ROUGE-2 | ROUGE-L |
|---|---|---|---|
| TextRank (Mihalcea and Tarau, 2004) | 49.04% | - | - |
| LexRank (Erkan and Radev, 2011) | 37.36% | - | - |
| AttSum (Cao et al., 2016) | 43.92% | 11.55% | - |
| SummaRuNNer (Nallapati et al., 2017) | 42% | 16.9% | 34.1% |
| Classify/Select (Nallapati et al., 2016b) | 42.2% | 16.8% | 35% |
| NeuSum (Cheng and Lapata, 2016) | 42.2% | 17.3% | 34.8% |
| MatchSum (Zhong et al., 2020) | 44.22% | 20.62% | 40.38% |

Table 2.2: An overview of the state of the art extractive summarization methods. The first three models used DUC' 2002, 2004 and 2007 respectively, while the others used CNN/Daily Mail (AV).

## 2.4 Abstractive summarization

Abstractive Summarization is the task of generating short and concise summaries that capture the salient ideas of the source text. The generated summaries potentially contain new phrases and sentences that do not appear in the source text. This Section presents a comprehensive review of the various works performed in abstractive summarization field.

### 2.4.1 Early approaches

Early approaches to abstractive summarization include **sentence compression** (Cohn and Lapata, 2014) that aims to create a grammatical summary of a given sentence. Earlier, Barzilay and McKeown (2005) and Filippova and Strube (2008), introduced sentence fusion, which involves using bottom-up local multisequence alignment to identify phrases conveying similar information and statistical generation to combine common phrases into a sentence (Lin and Ng, 2019). Tanaka et al. (2009), introduced **sentence revision**, a method that generates sentences not found in the input document and synthesizes information across sentences.

These approaches offered little improvement over extractive methods and motivated the development of fully abstractive methods that usually contain the three following tasks, performed in pipeline fashion:

- **Information Extraction** aims to extract important information from the original document.

- **Content Selection** aims to select a subset of the candidate phrases extracted in the previous step for inclusion in the final summary.

- **Surface Realization** aims to combine the candidates using grammatical/ syntactic rules to generate a summary.

### 2.4.2 Graph-based methods

Another approach for abstractive summarization are Graph-based Methods, where graphs are the means to implement the three subtasks mentioned above. Graphs are chosen due to their expressiveness (Greenbacker, 2011) as they facilitate the extraction of the complex and abstract relations between the concepts that exist in the text. Li et al. (2016b) used event semantic link networks (ESLNs) for joint information extraction and content selection. ESLNs are capable of providing an abstract representation of the text, with each node corresponding to a particular event mentioned in the input text document and each edge encoding the semantic relation between the corresponding events. Mehdad et al. (2014) proposed the use of entailment graphs for content selection by detecting redundant sentences. In more detail, if two sentences have the same meaning, the less informative of them will be removed and if both have some parts that do not overlap with the other, none of them will be removed (Lin and Ng, 2019).

### 2.4.3 Template-based methods

Another approach for abstractive summarization are Template-based Methods that are based on the observation that human summaries of a given type have very similar sentence structures that can be learned from the human summaries in the training set and encoded as templates (Lin and Ng, 2019). Given a source document, a summary can be generated by filling the slots in the best fitted templates learned for this type of documents. Oya et al. (2014) introduced a robust template-based method for meeting summarization. At first, a template is being generated from a sentence of each human summary in the training set by replacing each Noun Phrase (NP) in the sentence with a blank slot that is labeled with the hypernym of the NP's head using WordNet. Next, the important phrases for each topic segment of the source document are extracted and labeled with their hypernyms. Finally, the templates with the highest similarity with each topic are being selected, as both the selected phrases and the most similar template have hypernym labels, candidate summary sentences can be generated by filling each template with matching labels. Here, the generation of a large number of sentences for each topic segment is very possible, so a sentence ranker is trained to rank the sentences in each segment with the highest ranked sentences for each topic are being selected for inclusion in the summary.

### 2.4.4 Neural approaches

In contrast with classical methods to abstractive summarization where information extraction, content selection, and surface realization are being applied, Neural Methods offer an end-to-end approach, learning how to abstract from the source document and generate the corresponding summary in one network (Lin and Ng, 2019). Rush et al. (2015) introduced the application of neural machine translation to abstractive summarization, sparking a novel way of building abstractive summarizers. After this work, neural methods have become the trend in building abstractive summarizers. These methods offer less control over what is being learned and how information is encoded, in comparison with classical methods.

Recent work on neural abstractive summarization mostly uses sequence to sequence (seq2seq) models, that apply the encoder-decoder architecture (Sutskever et al., 2014), that is composed of an encoder that encodes sentences as a list of fixed-length vector representations, each of which captures a word and its surrounding context, and a decoder that outputs a summary based on the encoded vectors (Lin and Ng, 2019).

The **Encoding** part focuses on capturing information relevant to summary generation. **Preprocessing** the input sentences is one of the two key steps of encoding. Most models use word-based representation, except the ones that work on languages like Chinese for which character-based representation might be a better alternative as it can avoid errors introduced by word segmentation (Chang et al., 2018). Also, the use of word vectors pre-trained on large corpora via word2vec (Mikolov et al., 2013) or GloVe (Pennington et al., 2014) is quite common, while fewer models learn the word embeddings during training (See et al., 2017).

Long documents are difficult to be encoded without the loss of important information, so the need to compress an input document into a more compact representation emerged. This was solved by leveraging extractive methods to select representative sentences (See et al., 2017; Hsu et al., 2018; Lebanoff et al., 2018). An effort to improve abstractive summarization was made, by adding background knowledge extracted from verified knowledge bases. Amplayo et al. (2018) extracted additional knowledge about the entities in the source document and used the resulting external knowledge as a guide to the decoder, in order to generate better summaries.

The second step of encoding is the **Encoder Selection**, that aims to learn a better abstract representation of the input document. CNN encoders with feed-forward Neural Network decoders, are replaced with RNNs in recent works, with the main reason to be the lack of CNNs' ability to process long sequences (Chopra et al., 2016; Nallapati et al., 2016a). To tackle this, researchers started applying LSTMs (Hochreiter and Schmidhuber, 1997) or GRUs (Chung et al., 2014) that have been shown as a better alternative to LSTMs due to their less parameter requirements and their speed in training, by performing equally good with LSTMs.

The most common practice for the **Decoding** part is the use of an RNN. Decoders generate each word in the output sequence given two sources of information. The

first one is the Context Vector, that is the encoded representation of the source document provided by the encoder, and the second one is the Generated Sequence, which is the word or sequence of words that is already generated. The decoder produces a vector matching the size of the vocabulary, which is turned into a distribution over the vocabulary using a softmax layer (Lin and Ng, 2019). Given this distribution, the most probable word is being generated or, more commonly, the k-best paths up to this step are identified through a beam search (See et al., 2017; Rush et al., 2015; Chopra et al., 2016; Nallapati et al., 2016a; Paulus et al., 2017).

Several attempts have been made to improve the encoder-decoder framework for abstractive summarization. The concept of **Attention** is motivated by the fact that some words or phrases are more important than others in a document, and such words or phrases should be given more chances in appearing in the summary. Attention is applied to identify such words or phrases by feeding the decoder with an extra vector (context vector) that encodes important phrases (Bahdanau et al., 2014) while de-emphasizing the unimportant information. One can employ sentence-level or word-level attention. This way, the resulting neural model may retrieve important information at different levels of a document (Nallapati et al., 2016a; Luong et al., 2015; Tan et al., 2017; Cohan et al., 2018).

Despite its benefits, Attention may cause over-focusing on certain phrases, which leads to redundancy in the generated summary. Nema et al. (2017) introduced the concept of **Distraction** to tackle this issue. The general idea is to employ a constraint to reduce the probability of repeated content or the weight associated with that content. Chen et al. (2016) showed that distraction can be applied to the context vector, the attention weight vectors, and decoding, although the application of distraction is not limited to these three places (Lin and Ng, 2019). While this concept was introduced as "Distraction", some researchers refer to it as **Coverage** (See et al., 2017). Coverage originated in statistical machine translation (Koehn et al., 2007) and is used for neural machine translation (Tu et al., 2016). See et al. (2017) proposed a coverage loss that has additional penalty for term repetition in a way that more repetition implies less coverage.

Another attempt to improve the encoder-decoder framework for abstractive summarization was made by Vinyals et al. (2015) that proposed a **Pointer Network** which copies an element from the input text directly to the output, in order to tackle the issue of the inability of neural sequence models to generate rare or out-of-vocabulary (OOV) words. This mechanism has been proven to be very useful for generating summaries, but there exists a crucial limitation. The overuse of the pointer by the decoder might lead to very similar results with extractive methods. Song et al. (2018) used the copy mechanism by copying a word from the input to the summary if "it contains salient semantic content, or it serves a critical syntactic role in the source sentence". The syntactic label of each word, such as its part-of-speech tag and its depth in the associated dependency parse tree, is encoded by the encoder network (Lin and Ng, 2019).

In the recent work of Zhang et al. (2020a), authors propose pre-training large Transformer-based encoder-decoder models on massive text corpora with a new self supervised objective. In PEGASUS, important sentences are removed/masked from

an input document and are generated together as one output sequence from the remaining sentences, similar to an extractive summary. The best PEGASUS model was evaluated on 12 downstream summarization tasks spanning news, science, stories, instructions, emails, patents, and legislative bills. Experiments demonstrate it achieves state-of-the-art performance on all 12 downstream datasets measured by *ROUGE* scores. Some of their results are presented in Table 2.3 below.

| Dataset | ROUGE-1 | ROUGE-2 | ROUGE-L |
|---|---|---|---|
| CNN/Daily Mail | 44.16% | 21.56% | 41.3% |
| Gigaword | 39.65% | 20.47% | 36.76% |
| NEWSROOM | 45.98% | 34.20% | 42.18% |
| arXiv (Cohan et al., 2018) | 44.21% | 16.95% | 25.67% |

Table 2.3: PEGASUS state of the art *ROUGE* scores on some downstream datasets.

Liu and Lapata (2019) in their work, showcase how BERT can be applied in text summarization and propose a general framework for both extractive and abstractive models. They introduce a novel document-level encoder based on BERT, called BERTSUM, which is able to express the semantics of a document and obtain representations for its sentences. For abstractive summarization, they propose a new fine-tuning schedule which adopts different optimizers for the encoder and the decoder as a means of alleviating the mismatch between the two (the former is pretrained while the latter is not). Results (Table 2.4) on three different datasets showcase that their model achieves state-of-the-art results across the board both in extractive and abstractive settings. Figure 2.3 (Liu and Lapata, 2019) presents the original BERT architecture alongside with the one that is used for text summarization.

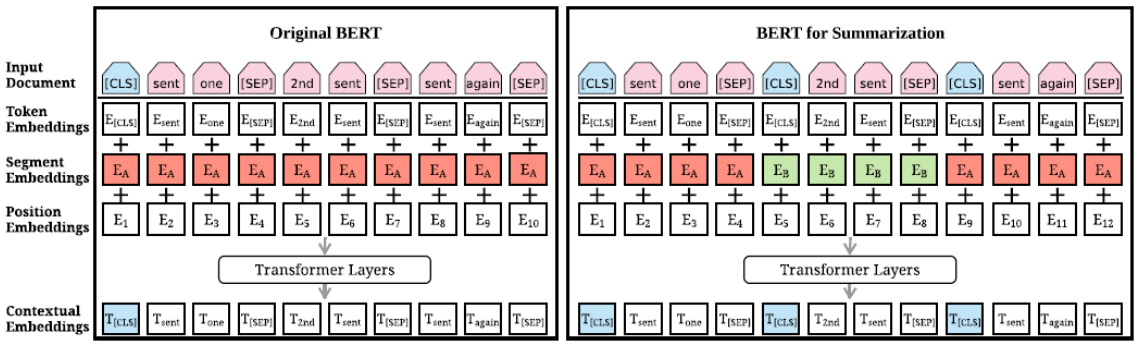


Figure 2.3: Architecture of the original BERT model (left) and BERTSUM (right). The sequence on top is the input document, followed by the summation of three kinds of embeddings for each token. The summed vectors are used as input embeddings to several bidirectional Transformer layers, generating contextual vectors for each token. BERTSUM extends BERT by inserting multiple [CLS] symbols (short for "classification") to learn sentence representations and using interval segmentation embeddings (illustrated in red and green color) to distinguish multiple sentences.

### 2.4.5 Reinforcement learning

Reinforcement Learning is an area of Machine Learning that allows machines and software agents to automatically determine the ideal behavior within a specific context, in order to maximize its performance while simple reward feedback is required for the agent to learn its behavior. Scientists have applied reinforcement learning techniques in NLP and text summarization in order to address some issues that could not be resolved with other techniques.

To start with, a solution to the problem of word/phrase repetition, that was encountered by Nallapati et al. (2016a), where attentional encoder-decoder models produced abnormal summaries that consisted of repeated phrases, was given by Paulus et al. (2017) who presented an objective function that maximizes the *ROUGE* metric. In more detail, they implemented a mixed training objective function that combines the teacher forcing algorithm and a function that maximizes the *ROUGE* metric. However, the optimization of *ROUGE* could not guarantee better quality, readability, coherence etc. due to *ROUGE's* limitations.

In their evaluation, the authors tried different models that varied between separately using Machine Learning(ML) and Reinforcement Learning(RL), to combining both of these functions. The RL algorithm performed the highest scores, with a $ROUGE_1 = 41.16\%$ and $ROUGE_L = 39.08\%$ compared to the other variations, while the ML+RL model scored the highest $ROUGE_2 = 15.82\%$, on the CNN/Daily Mail test dataset.

The authors performed human evaluation to ensure that the increase in *ROUGE* scores is also followed by an increase in human readability and quality, in order to find out whether the ML+RL training objective did improve readability compared to RL. To perform this evaluation, 100 test examples were randomly selected from the CNN/Daily Mail dataset and for each one of them, they showed the original article, the ground truth summary as well as summaries generated by different models side by side to a human evaluator (5 in total) who had to assign two scores from 1 to 10 to each summary, one for relevance and one for readability. Results showed that, despite the RL algorithm achieving the highest $ROUGE_1$ and $ROUGE_L$ score, it produced the least readable (4.18) and relevant (6.32) summaries. The most common readability issue observed in the RL results was that there were a lot of short and truncated sentences towards the end of the summary. On the other hand, the RL+ML summaries obtain the highest readability and relevance scores among all models achieving a readability and relevance of respectively 7.04 and 7.45 respectively, while the ML model had a 6.76 and 7.14, hence, solving the readability issues of the RL model while also having a higher *ROUGE* score than ML. This, according to the authors, demonstrates the usefulness and value of the RL+ML training method for abstractive summarization.

Stiennon et al. (2020) proposed a model that learns to summarize from human feedback. In this work researchers collected a large high quality dataset of human comparisons between summaries, trained a model to predict the human-preferred summary, and used that model as a reward to fine-tune a summarization policy using reinforcement learning. They applied this method to a version of the TL;DR dataset

of Reddit posts (Völske et al., 2017) and noticed that their models significantly outperform both human reference summaries and much larger models fine-tuned with supervised learning alone. Their models also transfer to CNN/Daily Mail news articles, producing summaries nearly as good as the human reference without any news-specific fine-tuning. The main limitation of this work, is the time and cost required to produce the final models. Notably, fine-tuning their model with RL required approximately 320 GPU-days, the training set took thousands of labeler hours and required significant researcher time to ensure quality.

### 2.4.6 Hybrid methods

The Hybrid approaches combine both the extractive and abstractive methods. The typical architecture of a hybrid text summarizer consists of the following phases (Bhat et al., 2018; Lloret et al., 2013; El-Kassas et al., 2021):

1. Pre-Processing,
2. Sentence extraction (extractive phase): extract the key sentences from the input document (Wang et al., 2017),
3. Summary generation (abstractive phase): generate the final abstractive summary by applying the abstractive methods and techniques on the extracted sentences from the first phase,
4. Post-Processing: to ensure that the generated sentences are valid, some general rules need to be defined like (Lloret et al., 2013):
    - The minimal length for a sentence must be 3 words (i.e.subject + verb + object).
    - Every sentence must contain a verb.
    - The sentence should not end with an article (e.g. "a", and "the"), a preposition (e.g. "of"), a conjunction (e.g. "and"), nor an interrogative word (e.g. "who").

Hybrid text summarization systems provide a combination of the advantages of both extractive and abstractive approaches which are complementary to each other, and the overall performance of summarization is improved (Wang et al., 2017). However, they lead to the generation of lower quality abstractive summaries than the pure abstractive approach because the generated summaries depend on the extracts, instead of the original text (El-Kassas et al., 2021). Generally, hybrid text summarization approaches are divided into **Extractive to Abstractive** and **Extractive to shallow Abstractive** methods.

**Extractive to Abstractive** methods start by using one of the extractive methods then they use one of the abstractive text summarization methods which is applied to the extracted sentences. Wang et al. (2017) proposed a hybrid system for long text summarization, called "EA-LTS". This system consists of two phases: 1) the extraction phase, that uses a graph model to extract the key sentences, and 2) the

abstraction phase that constructs an RNN based encoder-decoder and a pointer and attention mechanism to generate summaries. Liu et al. (2018) used a hybrid approach to generate Wikipedia pages by summarizing long sequences. Since the amount of text in reference documents could be very large, it was infeasible to train an end-to-end abstractive model given the memory constraints. Hence, they coarsely select a subset of the input using extractive summarization. The second stage involved training an abstractive model that generates the Wikipedia text while conditioning on the extraction. This two-stage process is inspired by how humans might summarize multiple long documents: first highlight pertinent information, then conditionally generate the summary based on the highlights.

Chen and Bansal (2018) proposed a model that selects and extracts important sentences and then rewrites them abstractively. First, sentences are represented using a temporal convolutional model and words are converted to a distributed vector representation by using word embeddings. Sequences of word vectors are fed to the model to capture the dependencies of nearby words. The sentence selection is done by training a pointer network based on a set of features (Vinyals et al., 2015). Then, an abstractive model compresses and paraphrases these sentences in order to create a concise summary sentence. The encoder-decoder structure by Bahdanau et al. (2014) is used and a copy mechanism (See et al., 2017) is added to help out with out-of-vocabulary words. In this work, authors use a reinforced learning technique based on *ROUGE* score in a way that model encourages the cases when the rewritten sentence scores high in *ROUGE* and discourages the cases when the rewritten sentence scores low in *ROUGE*. The "End-Of-Extraction" parameter is being added during the learning phase, which rewards the agent for finding the correct number of sentences for the summary. Their results showed improvement both in *ROUGE* score and in relevance and readablity from human users, comparing their model and the pointer model.

**Extractive to shallow Abstractive** methods start by using one of the extractive methods then they use a shallow abstractive text summarization method that applies one or more of the following techniques to the extracted sentences: information compression, information fusion (Lloret et al., 2013), synonym replacement (Patil et al., 2014), etc. Bhat et al. (2018) proposed a single-document hybrid ATS system called "SumItUp" which consists of two phases. At first, an extractive sentence selection that uses some statistical features (sentence length, sentence position, TF-IDF, noun phrase and verb phrase, proper noun, aggregate cosine similarity, and cue-phrases) and a semantic feature (emotion described by text) is being applied to generate the summary. Next, the extracted sentences are fed to a language generator to convert the extractive summary to the abstractive summary. To retain the original sequence, sentences are reordered based on their initial index.

In conclusion, the hybrid summarization approach is a promising research direction. Mahajani et al. (2019) recommend researchers to propose hybrid Automatic Text Summarization systems in order to benefit from the advantages of both extractive and abstractive approaches.

### 2.4.7 State of the art

Table 2.4 presents the most important methods from related work in abstractive summarization, including *ROUGE* scores for each one of them. Methods marked with a star (*) use the anonymized version of the CNN/Daily Mail corpus, while the others use the non-anonymized one.

| Method | ROUGE-1 | ROUGE-2 | ROUGE-L |
|---|---|---|---|
| Fast Abstractive Rewriting (Chen and Bansal, 2018) | 42.6% | 18.8% | 38.5% |
| Pointer-Generator (See et al., 2017) | 39.53% | 17.28% | 36.38% |
| Facebook: Controllable Abstractive Summarization (Fan et al., 2018) | 40.38% | 17.44% | 37.15% |
| Fast Abstractive Rewriting* (Chen and Bansal, 2018) | 39.66% | 15.85% | 37.44% |
| SalesForce: Reinforced Learning* (Paulus et al., 2017) | 39.87% | 15.82% | 36.9% |
| Facebook: Controllable Abstractive Summarization* (Fan et al., 2018) | 39.06% | 15.38% | 35.77% |
| BERTSUM_ABS* (Liu and Lapata, 2019) | 41.72% | 19.39% | 38.76% |

Table 2.4: An overview of the state of the art abstractive summarization methods.

# 3 Methods

This Chapter describes the methodology for this research. In Section 3.1 the general goal of our summaries will be discussed as well as the main criteria that will lead to the choice of the baseline and main models (Sections 3.2 and 3.3 respectively) that will be used. Moreover, the datasets and the evaluation methods will be analyzed alongside with a specific analysis for the chosen models in Section 3.4.

## 3.1 Goal of a summary

Before choosing the specific models that we will work with, it is quite important to define the stakeholders for this research. This determines the goal of our summary, and thus the main points that it needs to convey. Generally, a summary is meant to inform the reader—who has not read the text—of what the text is about. It describes its purpose or main idea, and gives an overview of supporting arguments that develop that idea or an overview of the most salient facts from a news article. The reader will then know if he or she will find it useful and want to read it or be able to get informed in a faster manner, without having to read the whole article. Another type of reader could be, in the case of a professional setting, for example a newspaper director that needs to decide if a reporter should be sent to an event, take the job, etc. So, the generated summaries should retain core facts found in the original article without losing any key point of it. Moreover, generated summaries should follow grammatical rules in order to be readable, coherent, faithful to the original text, without missing the fact that it must be concise. Summaries are much shorter than the original material—a general rule is that they should ideally be no more than 20% to 25% of the length of the original article.

## 3.2 Choosing baseline models

A baseline is a method that uses heuristics, simple statistics, randomness, or machine learning to create predictions for a dataset. One can use these predictions to measure the baseline's performance (e.g. accuracy). This metric will then become the one that we will compare any other, more advanced, text summarization algorithm against. Thus, our baselines are the basis we need for comparison of our results. This Section presents a high-level description of the baselines that are chosen for this study alongside with the main reasons they are chosen.

What we are looking for, are algorithms that encounter the problem of text summarization with a more simple way than our main two abstractive models. It seems

like extractive methods are simpler by their nature, compared with abstractive or hybrid methods. To begin with, a very strong and commonly used extractive baseline that was introduced by Nallapati et al. (2017) is the **LEAD-3 benchmark**, which generates summaries by extracting the first three sentences of the source document. The main reason why we chose this particular method, stems from the fact that despite being simple, it scores high on the *ROUGE* metric and is competitive with state-of-the-art systems. Furthermore, one must be curious about how such a naive approach to text summarization gives a good summary, according to *ROUGE*. Thus, the human evaluation experiment will showcase whether users prefer this over more complicated abstractive methods.

The **LEAD-3** baseline is one of the simplest ways to approach text summarization. So, we would like our next baseline to be a little "smarter" than LEAD-3. **TextRank** (see Section 2.3.1) (Mihalcea and Tarau, 2004) is an extractive and unsupervised text summarization technique that splits the original text into sentences, measures the similarities between sentence vectors and stores them in a matrix which is transformed to a graph with sentences as vertices and similarity scores as edges, for sentence rank calculation. Then, a certain number of top-ranked sentences form the final summary. This algorithm is chosen as our second baseline model, due to its simpleness and availability online, but mainly due to its suitability for a text summarization baseline. In this research we will use a 25% ratio to extract the most salient sentences of the document according to basic summarization "rules of thumb" (https://www.sps186.org/downloads/basic/279606/Good_Summaries.pdf) that indicate that the ideal length of the summary should be approximately equal to the 25% of the original text's size.

For our last baseline, we are looking for an extractive approach that uses more up-to-date techniques. As it has been discussed in Section 2.3 there are many algorithms that cover our needs, but judging according to automatic evaluation performance, code's availability, publication date and the dataset used, **MatchSum** (Zhong et al., 2020) is the one that stands out. This study works with a Siamese-BERT architecture and its performance is among the state-of-the-art models on CNN/Daily Mail dataset (44.41% in $ROUGE_1$) and the official repository is available publicly (https://github.com/maszhongming/MatchSum). Another work that was among our top choices is SummaRuNNer (Nallapati et al., 2017) but their implementation is not available, their model is being trained only on the Daily Mail dataset and the work of Zhong et al. (2020) is more recent which led to the later to be chosen.

## 3.3 Choosing main models

This Section discusses the decision around the two main models that will be used in this research. These methods are the core of this study as they will be the means to figure out what kind of summaries human users prefer. As it is explained before, we are looking for an abstractive text summarization model that uses an encoder-decoder attention architecture and a hybrid method that first extracts salient sentences from the text and then rewrites them abstractively. The main criteria under which this choice will be made are:

1. Performance on automatic evaluation metrics,
2. Human evaluation performance,
3. Code availability and documentation,
4. Dataset: CNN/Daily Mail,
5. Publication date.

Starting with the abstractive model, Section 2.4 discusses many different abstractive summarization models but since we are working on the CNN/Daily Mail dataset we limit our search space to the works that train and test their model to this dataset. After discarding many models that were mentioned in previous Sections due to them not satisfying the aforementioned criteria, the final two models that stood out are **Get to The Point** (See et al., 2017) and **BERTSUM_ABS** (Liu and Lapata, 2019). Both of these studies, perform very good on the *ROUGE* metric on CNN/Daily Mail but BERTSUM_ABS outperforms the Pointer-Generator Network approach. Also Lapata & Liu's work is more recent. As for their implementations, both offer their code publicly and their documenation is equally good. After evaluating these models, it was decided that the one that will be used in this research is **BERTSUM_ABS** (Liu and Lapata, 2019) since it seems more suitable according to our criteria.

As for the hybrid method, things are pretty straightforward since there is no previous work that outperforms the **Fast Abstractive Summarization with Reinforce-Selected Sentence Rewriting** (Chen and Bansal, 2018) model in any of the criteria mentioned above. Their work, besides being one of the most recent ones on hybrid text summarization it also has impressive *ROUGE* score results on the CNN/Daily Mail dataset. Moreover, their implementation is available (`https://github.com/ChenRocks/fast_abs_rl`) and their documentation is comprehensive. Thus, this will be the final model that will be used in this research.

## 3.4 CNN/Daily Mail dataset & outputs' exploration

In this research, we use the **CNN/Daily Mail** dataset first proposed by Hermann et al. (2015) for reading comprehension task. This dataset has been modified for summarization by Nallapati et al. (2017) and has been widely used in automatic text summary tasks, in recent years, due to its large data volume and long text content. The standard split of the dataset contains 287,227 documents for training, 13,368 documents for validation, and 11,490 documents for testing. On average, there are about 28 sentences per document in the training set (Hermann et al., 2015). Note that the original release of this dataset by Hermann et al. (2015) is an anonymized version, where the named entities are anonymized and treated as a single word in the evaluation n-gram matching. On the other hand, See et al. (2017) proposed to use the non-anonymized, original-text version of the dataset, which is used in this

research. An interesting observation on the dataset is that the Daily Mail part has around 2 times as many training documents but around 10 times as many validation and test documents. This means that CNN is underrepresented in the test data and the *ROUGE* results might be affected if there is a difference in performance between the two sources. Figure 3.1 showcases some basic statistics of the CNN/Daily Mail dataset as introduced by Hermann et al. (2015). "#months" row showcases the number of months that the researchers needed to collect the respective documents while all other rows of this table are self-explanatory.

| | **CNN** | | | **Daily Mail** | | |
|---|---|---|---|---|---|---|
| | train | valid | test | train | valid | test |
| # months | 95 | 1 | 1 | 56 | 1 | 1 |
| # documents | 90,266 | 1,220 | 1,093 | 196,961 | 12,148 | 10,397 |
| Avg # tokens | 762 | 763 | 716 | 813 | 774 | 780 |
| Vocab size | 118,497 | | | 208,045 | | |

Figure 3.1: CNN/Daily Mail basic statistics.

### 3.4.1 Output abstractiveness

Following the approach of See et al. (2017), we compute an abstractiveness score in order to measure the novelty level of the two main models. We calculate this score as the ratio of novel word $n$-grams in the generated summary that are not present in the original text of the input article. Moreover we examine the ratio of our models against the respective reference summaries, since the reference summaries are being treated as the target variable by both models, and we also measure the abstractiveness of the reference summaries against the original articles. Since punctuation was sometimes missing in the CNN/Daily Mail dataset, all symbols/punctuation were removed for computing the abstractiveness score according to `https://www.programiz.com/python-programming/examples/remove-punctuation`.

The formula that was used to calculate the abstractiveness ratios is presented below. Novel $n$-grams of the output summary are the ones that do not exist in the respective article or reference summary. The denominator represents the total number of $n$-grams in the output summary.

$$\text{Abstractiveness Ratio} = \frac{\#\text{novel n-grams in output}}{\#\text{n-grams in output}}$$

Moreover, an investigation of the cases where $n$-grams in the generated summaries do not exist neither in the article nor in the reference summary was performed. For these cases we present some interesting examples with the exact completely novel words that were generated by the models in Appendix B.

## 3.5 Human evaluation setup

To determine which of the models described in Section 3.3 produces useful, accurate and more natural summaries, we use *ROUGE* as our baseline metric and focus on the results of the human evaluation experiment. This Section discusses the experimental setup.

Following previous work, this human experiment will be asking human participants to rank the summaries that were generated by the three baseline and two main models of this research. The criteria under which participants will determine which summaries they prefer are **Readability** and **Relevance**. Readability assesses the fluency, grammaticality and the length of the summary. Relevance is based on whether the summary contains all important information of the original document and whether it avoids generating repeated and redundant information.

Each subject will be given 50 articles and their respective summaries that were automatically generated by **LEAD-3**, **TextRank**, **MatchSum**, **BERTSUM_ABS** and **Fast Abstractive Rewriting**. These instances will be chosen randomly from the CNN/Daily Mail test set. 35 articles will be the same for every participant and the remaining 15 will be different for each participant. For each instance, participants will be asked to give scores (0-10) to six summaries of the same article (5 automatically generated summaries + 1 reference summary) according to Readability and then Relevance. After participants complete the survey, they will be asked questions about their experience through optional online meetings. In these meetings, human participants will be interviewed for about 15-30 minutes after they have completed the survey. As a first step, participants will be free to express their thoughts and comments on the experiment. Next, they will be called to answer specific questions in which they will have to state which summary they liked the best and why, if they found anything strange in the experiment and if they believe that sentence extraction is a good way to generate summaries.

Since reading English news articles is a task that is being performed daily by people all over the world, this survey will not involve only native English speakers, but people with a decent knowledge of the English language.

What follows is the protocol of this survey. That is the first thing that participants will see and will give them general instructions for the evaluation task. Figure 3.2 presents the "Edit your response" link, an option that is given to participants that do not have the time to complete the survey in one session. This link should be saved, and allows participants to continue at a later time.

### Survey Protocol

Welcome to the survey for my MSc Thesis Project "A comparison between Abstractive and Semi-Abstractive methods for text summarization on the CNN/Daily Mail dataset".

In this survey, 50 original news articles that were randomly selected from a pool of articles that were collected from CNN and Daily Mail feed, will be presented. After reading each article, 6 candidate summaries will be presented for it and you are asked to assign a score (0 is the worst and 10 is the best possible score, decimals are allowed) according to the following criteria:

Readability: assesses the fluency, coherence, grammaticality and the length of the summary.

Relevance: whether the summary contains all important information of the original article and whether it avoids generating repeated and redundant information.

Please read the article and all 6 summaries in full before assigning the scores.

IMPORTANT TIPS:

---

1) In some cases you might see names or phrases that should start with capital letters but they don't (e.g. "ioannis tsogias" instead of "Ioannis Tsogias"), or strange symbols like "$\langle q \rangle$" at the end of each sentence which represents the "." symbol. It is likely to see the token "[UNK]" instead of a word. Also some dots (".") might be missing from the end of a sentence (in that case you will see two words connected e.g. "..takengeorge.." which should be "..taken. George..") . Please, do not take these minor mistakes under consideration, this survey focuses on the content of each text and not such minor formatting details. Assume that everything is correct in terms of spelling and punctuation and evaluate only the content of each summary.

2) A text that does not follow grammatical rules like "go i super market to the" is something that should reduce the readability score for this particular summary.

3) You don't have to complete the survey in one session. But you have to be careful as everything can be lost if the following instructions will not be followed.

Let's say that you start filling in the survey and have to close it after you have read and evaluated the summaries of the first 10 articles out of the 50. In such a case, you can skip the rest of the articles, by clicking "Next" and submit the survey. After submitting, you will be guided to a page where you will be said that "Your response has been recorded". As you may see in the image below (Figure 3.2), you can choose to edit your response. Click that option and you will be guided to the survey that you submitted with all answers saved. In order to continue from the point you stopped, you should save the link of the "Edit your response" choice and visit it when you decide to continue filling in the survey.

---

An optional online meeting after having finished the survey would be very helpful for me. In this meeting you will be asked questions about your experience filling out the survey.

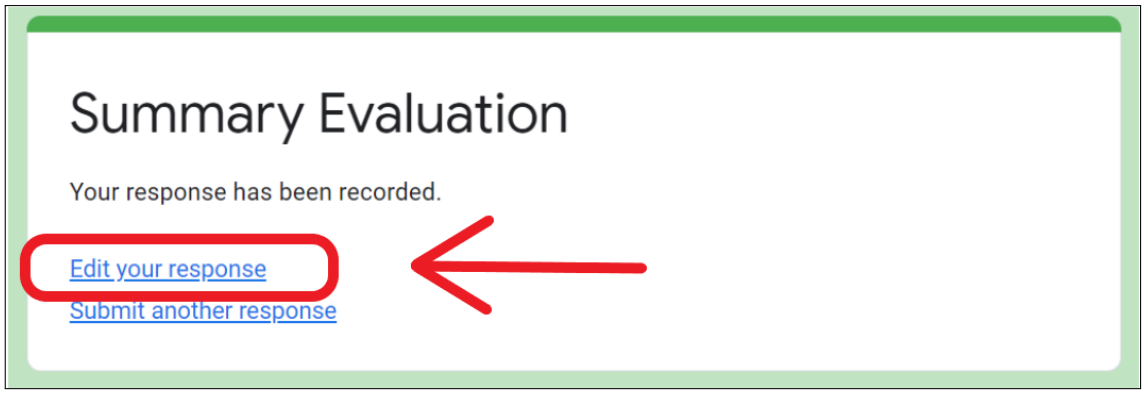


Figure 3.2: Participants that do not want to complete the survey in one session should save their progress by saving the "Edit your response link" in order to continue at a later time.

# 4 Results

This Chapter presents the results of the algorithms as described in Chapter 3 on the CNN/Daily Mail dataset alongside with comments and observations on them. We evaluated summarization quality automatically using full length $F_1$ $ROUGE$ score, following previous work (Lin, 2004b). In Section 4.1 we report unigram and bigram overlap ($ROUGE_1$ and $ROUGE_2$) as a means of assessing informativeness and the longest common subsequence ($ROUGE_L$) as a means of assessing fluency. Section 4.2 presents the results of the human evaluation experiment.

## 4.1 Automatic evaluation

As shown in the first part of Table 4.1, the results of the baseline methods are being listed. To begin with, both of **MatchSum** models seem to outperform the rest of the baselines by far, according to $ROUGE_1$, $ROUGE_2$ and $ROUGE_L$. When the encoder was changed to RoBERTa-base (Liu et al., 2019), the performance was further improved. According to Zhong et al. (2020), the small improvement here is because RoBERTa introduced 63 million English news articles during pretraining. The superior performance on this dataset demonstrates the effectiveness of their matching framework.

What is interesting here is that **MatchSum**, an extractive text summarization approach, besides beating all baselines, also achieves better results than our two main abstractive models. In this case two out of the three extractive baselines seem to work well for this dataset. Except **MatchSum**, **LEAD-3** also performs unexpectedly good, as its $ROUGE$ scores are almost equal with **Fast Abstractive Rewriting** according to $ROUGE_1$ and $ROUGE_2$, while their difference in $ROUGE_L$ is only equal to **2.17%**.

On the other hand, **TextRank** (ratio = 0.25) did not perform well and the comparison with the other models seems unfair since TextRank comes last by far, with the next best model (LEAD-3) to surpass it by **15.46%**, **8.01%** and **9.06%** on $ROUGE_1$, $ROUGE_2$ and $ROUGE_L$ respectively. This poor performance of TextRank led us to investigate the ratio = 0.3 scenario but with no further success. The second version of the TextRank algorithm resulted worse than the first one. Although it scored slightly better in $ROUGE_2$, it is being outperformed by the first version in terms of $ROUGE_1$ and $ROUGE_L$.

| Method | ROUGE-1 | ROUGE-2 | ROUGE-L |
|---|---|---|---|
| LEAD-3 | 40.42% | 17.62% | 36.67% |
| TextRank ratio = 0.25 | 24.96% | 9.61% | 27.61% |
| TextRank ratio = 0.3 | 23.7% | 9.64% | 27.47% |
| MatchSum BERT-base | 44.22% | 20.62% | 40.38% |
| **MatchSum RoBERTa-base** | **44.41%** | **20.86%** | **40.55%** |
| Fast Abstractive Rewriting rnn-ext + abs + RL + rerank | 40.88% | 17.80% | 38.54% |
| Fast Abstractive Rewriting rnn-ext + abs + RL | 40.04% | 17.61% | 37.59% |
| Fast Abstractive Rewriting rnn-ext + abs | 38.38% | 16.12% | 36.04% |
| **BERTSUM_ABS** | **41.72%** | **19.39%** | **38.76%** |

Table 4.1: Results of the baseline and main methods on the CNN/Daily Mail dataset.

Moving on to the main models, as shown in the second part of Table 4.1, **BERTSUM_ABS** outperforms all three versions of the **Fast Abstractive Rewriting** model in all three different *ROUGE* metrics.

As mentioned in Chapter 2, Chen and Bansal (2018) proposed a hybrid approach that is capable of both extractive and abstractive (i.e., rewriting every sentence) summarization. They run experiments on separately trained extractor/abstractor (rnn-ext + abs) and the reinforced full model (rnn-ext + abs + RL) as well as the final reranking version (rnn-ext + abs + RL + rerank).

As one may see in Table 4.1, adding the RL feature improves the performance of the rnn-ext + abs model. Although the extract-then-abstract approach inherently will not generate repeating sentences like other neural decoders do, there might still be across-sentence redundancy because the abstractor is not aware of other extracted sentences when decoding one. Hence, Chen and Bansal (2018) incorporate a reranking strategy where they remove a few 'across-sentence' repetitions, via a simple reranking strategy: At sentence-level, they apply beam-search tri-gram avoidance (Paulus et al., 2017). They keep all $k$ sentence candidates generated by beam search, where $k$ is the size of the beam. Next, they rerank all $k^n$ combinations of the $n$ generated summary sentence beams. The summaries are reranked by the number of repeated N-grams, the smaller the better. They also apply the diverse decoding algorithm described in Li et al. (2016a) (which has almost no computation overhead) so as to get the above approach to produce useful diverse reranking lists. The improved ROUGE scores indicate that this successfully removes some remaining redundancies and hence produces more concise summaries. However, their final and most powerful model with the reranking feature does not manage to outperform **BERTSUM_ABS**.

This is the first step into answering the the main research question of this study

about which of the abstractive and hybrid methods produce the best summaries. **BERTSUM_ABS** outperformed Fast Abstractive Rewriting according to *ROUGE*, and **MatchSum's** performance was the best one among the baseline models. However, according to our hypothesis that *ROUGE* is not the optimal way to evaluate summaries, we will wait for the results of the human evaluation experiment to come up with a final answer.

### 4.1.1 Abstractiveness

Table 4.2 showcases the abstractiveness ratios of BERTSUM_ABS, compared with the original article and the reference summaries in 4 different settings where $n$ ranges between 1 and 4. Table 4.3 presents the same analysis for the Fast Abstractive Rewriting model. Both models' outputs seem to be using the same words as the original article, almost exclusively, since the ratio of 1-grams is equal to 1.45% and 0.96% respectively. The ratio increases as $n$ gets higher, which means that in terms or phrases, or even sentences, both models create novel texts with ratios equal to 29.13% and 33.64% respectively. The abstractiveness ratios of both models are significantly increased when compared with the reference summaries. Both models create summaries that use novel 1-grams (words) with a ratio higher than 50%, while reaching ratios higher than 90% when $n = 4$.

| **Comparison** | **1-grams** | **2-grams** | **3-grams** | **4-grams** |
|---|---|---|---|---|
| BERTSUM_ABS vs Article | 1.45% | 11.65% | 21.47% | 29.13% |
| BERTSUM_ABS vs Reference | 52.5% | 80.74% | 88.88% | 92.41% |

Table 4.2: Abstractiveness ratios for BERTSUM_ABS.

| **Comparison** | **1-grams** | **2-grams** | **3-grams** | **4-grams** |
|---|---|---|---|---|
| F.A.R. vs Article | 0.96% | 12.33% | 23.74% | 33.64% |
| F.A.R. vs Reference | 53.48% | 82.98% | 91.11% | 94.45% |

Table 4.3: Abstractiveness ratios for Fast Abstractive Rewriting model.

Table 4.4 presents the analysis of the abstractiveness ratios for the reference summaries against the original articles. Here, we observe that the reference summaries use words from the original article with a ratio equal to 12.56%, which increases in a parallel manner with $n$. As presented in the Table, the reference summaries use novel 4-grams with a ratio of 81.09%. However, the fact that reference summaries use novel words only at 12.56% of the time, means that they are not that much abstractive. Thus, the nature of this particular dataset is not compatible with training abstractive models by using reference summaries that mostly copy words from the original article and are indeed more extractive than abstractive. This can also be concluded by looking the abstractiveness ratios of BERTSUM_ABS and Fast Abstractive Rewriting which showcase that their summaries mostly tend to copy words from the text and rarely create novel ones.

Unfortunately, not all previous work reports on abstractiveness for their models in order to be able to compare our results. But, Chen and Bansal (2018) do compute

abstractiveness scores for their model which is not significantly different from our results. The Fast Abstractive rewriting model's abstractiveness scores (compared with the original article) in their analysis correspond to [0.3% 10.0% 21.7% 31.6%] while scores for the Reference summaries are [10.8% 47.5% 68.2% 78.2%]. The reader can compare these vectors with the first row of Table 4.3 and Table 4.4 respectively and see that there are not significant differences between the two analyses. Furthermore, Liu and Lapata (2019) also report on abstractiveness but only for the bigram setting for the Reference summaries in the CNN/Daily Mail dataset, giving it a score of 52.9%, which differs both from our and Chen and Bansal (2018) analyses, but not significantly.

| Comparison | 1-grams | 2-grams | 3-grams | 4-grams |
|---|---|---|---|---|
| Reference vs Article | 12.56% | 51.45% | 71.63% | 81.09% |

Table 4.4: Abstractiveness ratios for the comparison between the original articles and their reference summaries.

Tables 4.5 and 4.6 showcase both models' abstractiveness ratios for the cases where $n$-grams in the generated summaries do not exist neither in the article nor in the reference summary.

| Comparison | 1-grams | 2-grams | 3-grams | 4-grams |
|---|---|---|---|---|
| BERTSUM_ABS vs Reference & Article | 1.21% | 10.81% | 20.55% | 28.36% |

Table 4.5: Abstractiveness ratios for the comparison between the outputs of BERTSUM_ABS against the original articles and their reference summaries.

| Comparison | 1-grams | 2-grams | 3-grams | 4-grams |
|---|---|---|---|---|
| F.A.R. vs Reference & Article | 0.88% | 11.9% | 23.28% | 33.27% |

Table 4.6: Abstractiveness ratios for the comparison between the outputs of F.A.R. against the original articles and their reference summaries.

## 4.2 Human evaluation

This Section presents the analysis of the results of the human evaluation experiment. It took subjects approximately 3 weeks to complete the survey and none of them did it in one session. Appendix C presents a full description of the properties of the participants in Table C.1.

To begin with, Table 4.7 showcases the mean Readability and Relevance scores for all 185 texts that were used in the experiment. In terms of readability, human subjects rated LEAD-3's summaries as the absolute best ones with a score of 8.36. What follows is MatchSum with 7.46, BERTSUM_ABS (7), TextRank (5.85) and Fast

Abstractive Rewriting (5.6). According to relevance scores, MatchSum produced the most relevant summaries to the original text with a score of 6.9, followed by LEAD-3 (6.86), TextRank (6.38), BERTSUM_ABS (6) and Fast Abstractive Rewriting (5.5). Furthermore, subjects were called to grade the reference summaries in terms of Readability and Relevance. Reference summaries got a score of 7.41 for their Readability and 6.68 for their Relevance, which is the 3rd best in both rankings if compared with every method.

| **Method** | **Readability** | **Relevance** |
|---|---|---|
| LEAD-3 | **8.36 ±1.22** | 6.86 ±1.50 |
| TextRank | 5.85 ±2.01 | 6.38 ±1.87 |
| MatchSum | 7.46 ±1.60 | **6.90 ±1.55** |
| BERTSUM_ABS | 7.00 ±1.95 | 6.00 ±1.89 |
| Fast Abstractive Rewriting | 5.60 ±2.05 | 5.50 ±2.00 |
| Reference | 7.41 ±2.12 | 6.68 ±2.35 |

Table 4.7: Results of the Human Evaluation Experiment on 185 articles of the CNN/Daily Mail dataset.

Table 4.7 presents the mean Readability and Relevance scores for the 35 instances that all 10 human subjects that participated in the experiment shared. These results indicate that the results of Table 4.8 are reliable since the ranking between the models does not change. The only difference is that Reference summaries are placed 4th, both in Readability and Relevance.

Results at Table 4.8 although reliable, are not able to showcase significant differences between the models. These results can not guarantee that the ranking that is presented is stable, due to standard deviation which is a measures of spread that summarizes a group of data by describing how spread out the values are. If the spread of values in the data set is large, the mean is not as representative of the data as if the spread of data is small. This is because a large spread indicates that there are probably large differences between individual data points. This analysis is not being applied to Table 4.7 as here, most texts were evaluated individually by each human subject. Only 35 texts were evaluated from everyone. Despite that, these results are a good indication of how human subjects preferred certain summaries more than others.

| **Method** | **Readability** | **Relevance** |
|---|---|---|
| LEAD-3 | **8.39 ±0.39** | 6.65 ±0.98 |
| TextRank | 6.16 ±0.92 | 6.34 ±0.99 |
| MatchSum | 7.77 ±0.72 | **7.02 ±1.15** |
| BERTSUM_ABS | 7.09 ±0.82 | 5.74 ±1.23 |
| Fast Abstractive Rewriting | 5.79 ±1.05 | 5.11 ±0.97 |
| Reference | 6.94 ±0.75 | 5.96 ±1.51 |

Table 4.8: Results of the Human Evaluation Experiment on 35 articles of the CNN/Daily Mail dataset.

In order to visualize our results we present the error bar graphs in Figures 4.1 and 4.2 that correspond to Tables 4.7 and 4.8 respectively. Error bars help us to see how spread the data are around the mean value and how accurately the mean value represents the data. More importantly, the standard deviation error bars are used here to get a sense for whether or not the differences between the performance of the models are significant. The less overlap two standard deviation error bars have with each other, the more probable it is for the difference between the variables they represent to be statistically significant. For example in Figure 4.2, there is no overlap between the readability error bars of LEAD-3 and TextRank, which is a strong indication that their difference is significant.

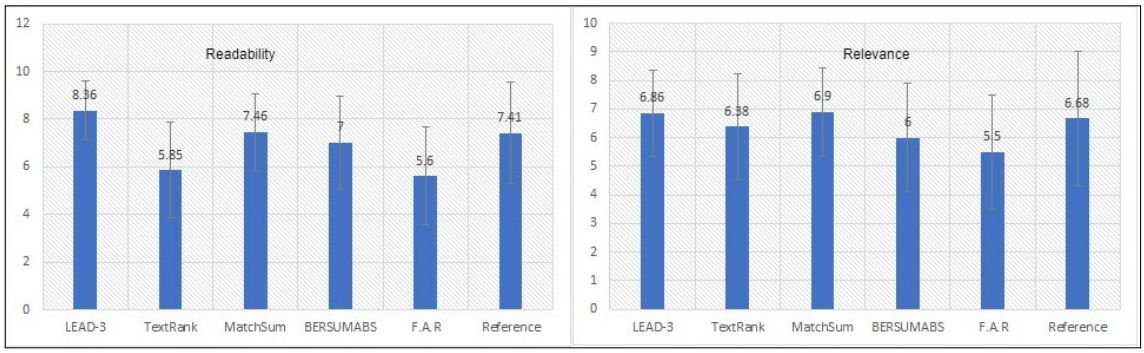


Figure 4.1: Standard deviation error bar graphs for the human evaluation results on all 185 instances.

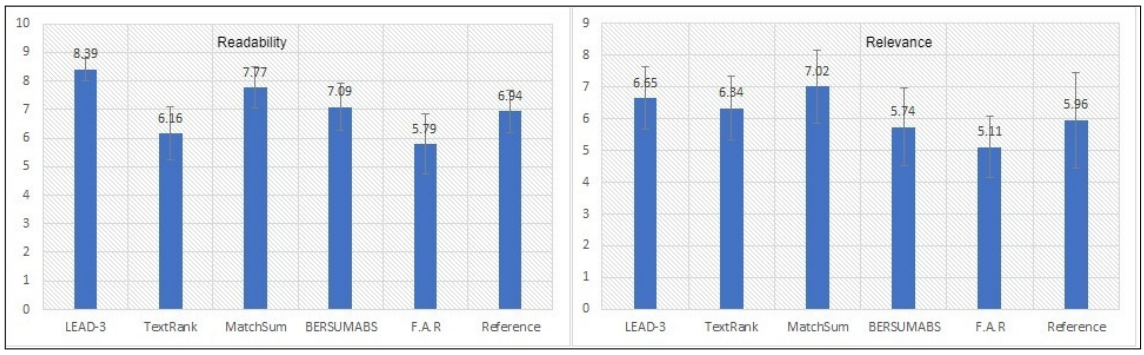


Figure 4.2: Standard deviation error bar graphs for the human evaluation results on the 35 instances that all participants shared.

### 4.2.1 T-test

In order to be sure of whether the differences between the performance of BERTSUM_ABS and Fast Abstractive Rewriting are statistically significant, we perform a T-test. The T-test is a test of a statistical significant difference between two groups. A "significant difference" means that the results that are seen are most likely not due to chance or sampling error. Below, Figure 4.3 showcases the formula of the T-test.

We perform the T-test for the results of the human evaluation experiment on 35 articles of the CNN/Daily Mail dataset that all participants shared, using "p value"

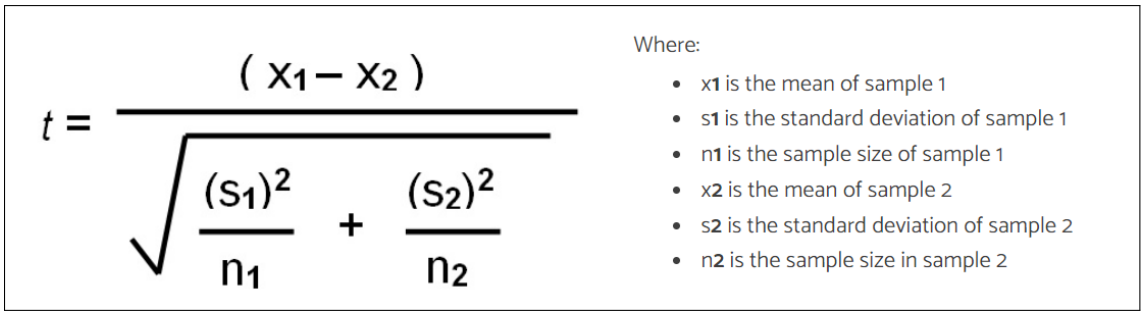


Figure 4.3: T-test formula. The t-Distribution table can be found here: https://home.ubalt.edu/ntsbarsh/business-stat/StatistialTables.pdf

of 0.05, our degree of freedom is equal to 34 (the degree of freedom equals to the size of sample minus 1). With these, we can determine the "Critical T-value" from the t-Distribution table which is equal to 2.032. Our Null Hypothesis is that there is not a significant difference between the performances of the two models. We calculate $t$ to be equal to 5.77 for Readability and 2.37 for Relevance. Both of these scores are greater than the critical T-value from the table, so we can conclude that the difference between the means for the two groups is significant in both cases. Thus, the null hypothesis is rejected and it is safe to conclude that the differences of the two models performance is statistically significant and that BERTSUM_ABS performed better than Fast Abstractive Rewriting both in terms of Readability and Relevance, according to human participants.

## 4.3 Correlation

After having collected and presented the results of the human evaluation experiment, we need to examine their relationship with the automatic evaluation results. Tables 4.9 - 4.13 present the Pearson correlation coefficient ($\rho$) between *ROUGE* scores and human evaluation. Each row and column represents a metric, e.g. in Table 4.9 L3_R1 represents the $ROUGE_1$ score for LEAD-3, L3_RDB the readability score, L3_RLV the relevance score and L3_Sum is a new variable that represents the mean value between Readability and Relevance.

It seems like *ROUGE* scores correlate with each other in all cases since $\rho > 0.5$ that makes these variables to be considered moderately correlated with each other, which was expected. The general looks of these correlation matrices indicate the automatic evaluation and human evaluation results have no relationship, since in all cases $\rho$ is very close to zero. The same thing applies to the case that *ROUGE* metrics are being compared with the new Sum variable.

Tables 4.14 and 4.15 present the correlation between Readability/Relevance scores with the articles' length for the baseline and main models respectively. Once again, these results indicate that there is no relationship between the presented variables and the length of the text since in most cases $\rho$ is very close to zero. This test was performed in order to investigate whether the text length affects the Readability or Relevance of the outputs.

| | L3_R1 | L3_R2 | L3_RL | L3_RDB | L3_RLV | L3_Sum |
|---|---|---|---|---|---|---|
| L3_R1 | X | 0.82 | 0.63 | 0.01 | -0.02 | -0.003 |
| L3_R2 | X | X | 0.87 | 0.06 | -0.02 | 0.02 |
| L3_RL | X | X | X | 0.07 | -0.03 | 0.015 |
| L3_RDB | X | X | X | X | 0.38 | 0.79 |
| L3_RLV | X | X | X | X | X | 0.86 |

Table 4.9: Correlation results for LEAD-3.

| | TR_R1 | TR_R2 | TR_RL | TR_RDB | TR_RLV | TR_Sum |
|---|---|---|---|---|---|---|
| TR_R1 | X | 0.82 | 0.61 | -0.01 | -0.12 | -0.07 |
| TR_R2 | X | X | 0.83 | -0.02 | -0.14 | -0.09 |
| TR_RL | X | X | X | -0.06 | -0.1 | -0.09 |
| TR_RDB | X | X | X | X | 0.5 | 0.89 |
| TR_RLV | X | X | X | X | X | 0.87 |

Table 4.10: Correlation results for TextRank.

| | MS_R1 | MS_R2 | MS_RL | MS_RDB | MS_RLV | MS_Sum |
|---|---|---|---|---|---|---|
| MS_R1 | X | 0.81 | 0.56 | -0.006 | 0.05 | 0.02 |
| MS_R2 | X | X | 0.83 | 0.06 | 0.09 | 0.08 |
| MS_RL | X | X | X | 0.06 | 0.06 | 0.07 |
| MS_RDB | X | X | X | X | 0.61 | 0.9 |
| MS_RLV | X | X | X | X | X | 0.89 |

Table 4.11: Correlation results for MatchSum.

| | BSA_R1 | BSA_R2 | BSA_RL | BSA_RDB | BSA_RLV | BSA_Sum |
|---|---|---|---|---|---|---|
| BSA_R1 | X | 0.81 | 0.56 | 0.01 | -0.004 | 0.003 |
| BSA_R2 | X | X | 0.84 | 0.07 | 0.03 | 0.06 |
| BSA_RL | X | X | X | 0.09 | 0.05 | 0.07 |
| BSA_RDB | X | X | X | X | 0.66 | 0.91 |
| BSA_RLV | X | X | X | X | X | 0.9 |

Table 4.12: Correlation results for BERTSUM_ABS.

| | FAR_R1 | FAR_R2 | FAR_RL | FAR_RDB | FAR_RLV | FAR_Sum |
|---|---|---|---|---|---|---|
| FAR_R1 | X | 0.82 | 0.52 | 0.04 | -0.009 | 0.01 |
| FAR_R2 | X | X | 0.76 | -0.001 | -0.01 | -0.01 |
| FAR_RL | X | X | X | 0.01 | 0.008 | 0.01 |
| FAR_RDB | X | X | X | X | 0.69 | 0.92 |
| FAR_RLV | X | X | X | X | X | 0.91 |

Table 4.13: Correlation results for Fast Abstractive Rewriting.

| | L3_RDBL | L3_RLV | TR_RDBL | TR_RLV | MS_RDBL | MS_RLV |
|---|---|---|---|---|---|---|
| Length | -0.05 | 0.01 | 0.03 | 0.08 | 0.0008 | 0.0002 |

Table 4.14: Correlation results for Baseline models' Readability and Relevance with the text length.

| | BSA_RDBL | BSA_RLV | FAR_RDBL | FAR_RLV | REF_RDBL | REF_RLV |
|---|---|---|---|---|---|---|
| Length | 0.11 | 0.07 | 0.13 | 0.12 | 0.1 | -0.03 |

Table 4.15: Correlation results for main models' and reference summary Readability and Relevance with the article's length.

# 5 Discussion

This Chapter presents the observations on both automatic and human evaluation results. After discussing around the automatic evaluation results and pointing out the limitations of *ROUGE* in Section 5.1 we will move on to the interpretation of human evaluation results, the performance of each model according to human participants and the correlation results (Section 5.2). Lastly, Section 5.3 discusses possible future work on the topic while suggesting some ideas for the improvement of the automatic evaluation process for automatically generated summaries.

## 5.1 Automatic evaluation results

The fact that **MatchSum** outperforms both of our main models and **LEAD-3** surpasses **Fast Abstractive Rewriting** in terms of $ROUGE_1$ and $ROUGE_2$ should be analyzed a bit further. What one would expect, is that more "intelligent" methods should generate better summaries than simple models like **LEAD-3**, but there might be some explanations for these observations.

Firstly, news articles tend to be structured with the most important information at the start; this partially explains the strength of the **LEAD-3** baseline. Indeed, as See et al. (2017) reports, using only the first 400 tokens (about 20 sentences) of the article during the training phase yielded significantly higher *ROUGE* scores than using the first 800 tokens.

Moreover, the nature of the task and the *ROUGE* metric make extractive approaches and the LEAD-3 baseline difficult to beat. The choice of content for the reference summaries is quite subjective as sometimes the sentences form a self-contained summary and other times they simply showcase a few interesting details from the article. Given that the articles contain 39 sentences on average, there are many equally valid ways to choose 3 or 4 highlights in this style. Abstraction introduces even more options (choice of phrasing), further decreasing the likelihood of matching the reference summary. For example, "smugglers profit from desperate migrants" is a valid alternative abstractive summary for the article in the box below, but it scores 0 *ROUGE* with respect to the reference summary. This inflexibility of *ROUGE* is sharpened by only having one reference summary, which has been shown to lower *ROUGE*'s reliability compared to multiple reference summaries (Lin, 2004a).

**Article**: smugglers lure arab and african migrants by offering discounts to get onto overcrowded ships if people bring more potential passengers, a cnn investigation has revealed.(...)
**Summary**: cnn investigation uncovers the business inside a human smuggling ring.

Due to the subjectivity of the task and thus the diversity of valid summaries, it seems that *ROUGE* rewards safe strategies such as selecting the first-appearing content, or preserving original phrasing. While the reference summaries do sometimes deviate from these techniques, those deviations are unpredictable enough that the safer strategy obtains higher *ROUGE* scores on average. This may explain why extractive systems tend to obtain higher *ROUGE* scores than abstractive, and even extractive systems, like MatchSum, do not significantly exceed the LEAD-3 baseline (See et al., 2017). To explore the issue further and be able to officially conclude whether *ROUGE* is an appropriate text summarization evaluation metric, we conducted a human evaluation experiment in order to examine the correlation between *ROUGE* results and human preferences so to confirm our initial intuition that *ROUGE* scores do not correlate with the human evaluation results and thus *ROUGE* is not suitable for the evaluation of summaries.

## 5.2 Human evaluation results

In order to continue, it should be useful to define the type of texts that are being analysed in this research. A news article discusses current or recent news of either general interest (i.e. daily newspapers) or of a specific topic (i.e. political or trade news magazines, club newsletters, or technology news websites). It can include accounts of eyewitnesses to the happening event. It can contain photographs, accounts, statistics, graphs, recollections, interviews, polls, debates on the topic, etc. Headlines can be used to focus the reader's attention on a particular (or main) part of the article. The writer can also give facts and detailed information following answers to general questions like who, what, when, where, why and how. Furthermore, news articles often contain the most salient information in the first few sentences in order to attract and engage readers to the text.

### LEAD-3

This fact gave LEAD-3 an advantage against the other models which was shown to be true as human readers gave the best scores to the summaries generated by this simple model. Most of the previous research highlights the fact that LEAD-3 is a very strong baseline, as it achieves high *ROUGE* scores and in this case it proved to be a really strong model as human participants preferred it against the other, more complex, models. However, the discussion with six out of the ten participants after the completion of the experiment showed that LEAD-3 does not produce good summaries, at least not the summaries that one would expect. LEAD-3 copies the first three sentences of the original text in order to produce summaries, which made participants incapable of giving low scores for these summaries' Readability. All participants were called to specify the reasons why every LEAD-3-generated summary did not get a perfect Readability score and most of them responded in a

similar fashion, stating that this is not the way a good summary should look like. Furthermore, human participants believed that in some cases, even the original article was not readable, which affected the performance of LEAD-3, and of course all extractive baselines. The issues with the readability of the articles were pointed out by all participants as the text was lowercased and in some cases punctuation was missing (an effort to make the texts look better was made, but I was not able to correct every mistake manually due to the lack of time and resources). Having these in mind, participants gave LEAD-3 good, but not perfect Readability scores. As for summaries' Relevance, as mentioned before, most articles include their most important information in the beginning, so in many cases the Relevance of LEAD-3 summaries was good. However, LEAD-3 was not capable of capturing the most salient information when the original article was long, which is not shown in the correlation analysis in Tables 4.14 and 4.15, but is a conclusion that was drawn after discussing with human participants. This does not mean that the correlation analysis is not consistent. Evaluating summaries is a subjective task as each human subject has its own beliefs and tastes. The majority of the participants that participated in the interview after completing the experiment pointed out that LEAD-3's outputs were summaries that could not get low scores on the chosen metrics as they were actually very accurate and efficient in most of the cases, but also stated that extraction should not be considered a good way to generate summaries. Thus, the reason why the fact that LEAD-3 was not able to capture salient information of long articles was not shown by the correlation analysis, might be that Readability and Relevance were not enough and that the interpretation of these measures was too strict.

#### TextRank

Moving on, TextRank did not perform as good as LEAD-3 as the extracted sentences were not smoothly connected and in some cases a big effort was needed in order to read and understand the whole text. Again, human participants faced the issue of having to rate a summary higher than they wanted to, due to its extractive nature. Relevance was quite good as the algorithm did capture most of the important information, but the lack of fluency of the generated summary made it look bad in terms of Readability and this is the main reason why it got low scores in both evaluation metrics.

#### MatchSum

On the other hand, MatchSum was the most balanced extractive baseline according to human participants. Most of the time it was able to capture the most important information of the original article and in a few cases it produced the exact same summary as LEAD-3. However, once again the sentences were not always connected smoothly with each other and participants believed that the extractive approaches, although useful, are not a good way to follow when generating summaries, which made them give lower scores to them. As mentioned in Section 2.3.4, Zhong et al. (2020) proposed Semantic Text Matching for extractive summarization as their main idea is that the generated summary should be the subset of the article's sentences

that is semantically closer to the article than any other candidate subset (summary), which seems to work as the discussion with human participants showed that it was the most stable model as it hardly ever got bad scores both for readability and relevance.

### Reference Summaries

Before analysing the performance of the two main models, an investigation of the quality of the reference summaries should be performed, as human participants did not rank them as high as expected, which can be problematic since these summaries are being treated as the target training variable for BERTSUM_ABS and Fast Abstractive Rewriting models. According to the discussion with the participants of the human evaluation experiment, the main issues that led them to give poor scores to the reference summaries are: 1) Sometimes the sentences were not smoothly connected, which is explained by the fact that reference summaries are not actual summaries, but rather highlights of the text. Human participants of course were not aware of that fact, but were able to recognize something problematic about them. 2) There were cases that reference summaries included information that was not present in the article, which of course is not something that should happen. A possible fix here could be to train and test only on instances with high *ROUGE* score, but this solution might not apply in different, more abstractive, datasets. 3) The size of these summaries was not following the rule of thumb that suggests summary's size to be approximately equal to 25% of the original text's size. Reference summaries were much smaller, which affected their relevance score as in the cases that the articles were long, small summaries were not able to capture all the salient information mentioned in the original text, a conclusion that was not shown in the correlation tests, but it was rather drawn from the discussion with human participants. As a result, these drawbacks of the reference summaries did also affect the performance of the two main models.

As mentioned in Section 2.1, Bommasani and Cardie (2020) in their recent study, state that data understanding is fundamentally important in natural language processing (NLP); for data-driven learning-based methods (e.g. neural networks), the quality of the training data bounds the quality of models learned using it. Therefore, understanding this data is necessary in order to ensure that models learn to perform a given task correctly. In their work, the researchers perform an intrinsic evaluation of summarization datasets and conclude that CNN/Daily Mail is one of the least abstractive datasets, noting that training learning-based systems (e.g. neural methods) using data with limited abstractivity implies the resulting summarizers will be limited in their ability to generate genuinely abstractive text. This is validated by empirical findings as both See et al. (2017) and Zhang et al. (2018) observe limited abstractivity in abstractive systems trained on CNN/Daily Mail which was shown in this project as well (Section 4.1.1). Also they concluded that this dataset is suboptimal for studying abstractive systems and that it is not a representative benchmark for summarization as a whole. Although our results are not able to prove this conclusion, they can be used as a strong indication that it is true.

#### BERTSUM_ABS

In general, BERTSUM_ABS had a good performance, its summaries were accurate most of the times, but there were cases that they included wrong facts, misplacing names and dates, as most of the human participants pointed out during the discussion session. Also, this model's outputs share the same drawbacks to the reference summaries as the size of BERTSUM_ABS' summaries was smaller than it should and the sentences were not smoothly connected. This is the only one amongst the models that are used in this research that conducted a Human Evaluation Experiment in a similar manner with us (as mentioned in Section 2.2.3), but not exactly the same. Liu and Lapata (2019) conducted human evaluation that included a question-answering (QA) paradigm which quantifies the degree to which summarization models retain key information from the document. In their work, human participants found BERTSUM_ABS more informative than LEAD-3, which can be compared to our relevance results. In this case, our analysis differs from theirs since the Relevance scores for LEAD-3 are higher than BERTSUM_ABS' (Tables 4.7 and 4.8), which can be explained as the tasks are not exactly the same and in both cases the tasks are rather subjective.

#### Fast Abstractive Rewriting

Surprisingly, the Fast Abstractive Rewriting model's outputs received the lowest scores both for Readability and Relevance by human participants. Even TextRank, which is the weakest of the baseline models, managed to surpass Fast Abstractive Rewriting in human evaluation. Despite the fact that the methodology of Fast Abstractive Rewriting was promising, since it follows the way that actual people write summaries (Mueller, 2008), not only didn't it produce the expected outcomes, but it was the worst out of all models analyzed in this research. Its generated summaries share all the disadvantages of Reference and BERTSUM_ABS summaries, but also in some cases Fast Abstractive Rewriting's outputs included weird words and terminated sentences that actually should not be stopped. Thus, the fluency of these summaries was poor while their Relevance was also low due to the many mistakes and wrong facts that were included in the summaries.

Therefore, comparing only our two main models with each other, we see that human users preferred the abstractive model more than the hybrid one. This is an indication, which can prove that our original intuition that miming human behaviour while summarizing texts, as Fast Abstractive Rewriting does, does not necessarily lead to high quality outputs. Thus, it is difficult to say for sure if hybrid models are better than abstractive models or not, however the results of this study favour the abstractive method.

### 5.2.1 Correlation

Intuitively, the fact that an automatically generated summary's n-grams overlap with some n-grams of the reference summary does not mean that this summary is actually good. This is the main reason why the correlation analysis was performed

(Tables 4.9 - 4.13), and showed that there is no relationship between participants' opinion and *ROUGE* metrics.

What can be drawn by this outcome, is that *ROUGE* can be a useful baseline evaluation metric, since it can be computed quickly and give us an idea of the quality of the summaries. However, without any other help, *ROUGE* is not enough to inform us whether the generated summaries are good or bad. For now, human evaluation might be the only way to help scientists make safe conclusions about the results since there is no other automatic way to evaluate all aspects of the automatically generated text, like its fluency, grammaticality, factual accuracy, completeness, length and abstractiveness. Not evaluating upon these measures in the human evaluation experiment was a decision that was made in order to keep the task simple. However, it seems like Readability and Relevance were too generic, which led to the confusion of human participants, but also were not able to evaluate upon all aforementioned aspects.

## 5.3 Future Work

Previous research, with the extensive use of *ROUGE* as an evaluation metric, has proven that it is the best one amongst all automatic evaluation metrics. However, scientists also claim that even if it is the most commonly used metric, its limitations do not allow researchers to make safe conclusions only according to it. Despite the fact that *ROUGE* can be useful to assess the quality of a summary, it mainly provides information regarding summary's informativeness, but there are more things to consider when evaluating a summary. So one might wonder of what should be used if not *ROUGE*. To the best of my knowledge, a proper automatic evaluation metric does not exist, at this moment. But this Section aims to give an overview of the main characteristics that should be considered when evaluating a summary and suggest a different way to approach the automatic evaluation issue, paving the way for future research to work on such a matter.

As mentioned before, the main characteristics that one needs to examine before evaluating an automatically generated abstractive summary are: text's fluency, grammaticality, factual accuracy, completeness, length and abstractiveness. Evaluating the quality of a summary under that many parameters can be very difficult, and its the nature of the human language that makes it difficult. The rules that dictate the passing of information using natural languages are not easy for computers to understand, and this is why Text Summarization and its evaluation process have not yet reached at perfect performance yet. So here, we will discuss about how close the scientific community is to the perfect evaluation metric, and what process could be followed in order to reach that level.

Calculating an abstractiveness score, or the length of the summary are rather easy, but important tasks as the length should not exceed the 1/3 of the text's size and the level of abstractiveness of the generated summary should be sufficient in order for it to be considered abstractive. As for grammaticality and fluency, the scientific community has achieved a lot, as the performance of the state-of-the-art models almost reach human-level performance in automatic grammatical error correction

(GEC). Ge et al. (2018), presented a state-of-the-art convolutional seq2seq model based GEC system that uses a fluency boost learning and inference mechanism. Fluency boost learning fully exploits both error corrected data and native data by generating diverse error-corrected sentence pairs during training, which benefits model learning and improves the performance over the base seq2seq model, while fluency boost inference utilizes the characteristic of GEC to progressively improve a sentence's fluency through round-way correction. Another, more recent work was presented by Omelianchuk et al. (2020), which suggests GECToR, a simple and efficient GEC sequence tagger using a Transformer encoder. Either of these works could be used in order to identify grammatical errors in automatically generated summaries.

Completeness corresponds to the summary comprising of all important information covered in input data, while factual accuracy refers to the level that information is being transferred from the article to the summary correctly. Generating fabricated facts has been a long-standing problem of abstractive summarization models, and has significantly limited their applicability in practice. Previous works about improving factual correctness only rely on human evaluations, which weakens the transparency and reproducibility. Zhang et al. (2020b) in their recent study examine how to evaluate factual correctness. After conducting a human study to thoroughly understand what affects factual correctness evaluations, they assess whether current automatic factual evaluation metrics are able to capture factual errors. The main focus of this work, is FactCCX (Kryscinski et al., 2020), which is a weakly-supervised, model-based approach for verifying factual consistency and identifying conflicts between source documents and a generated summary. Despite FactCCX outperforming other models trained on textual entailment and fact-checking data, it is still not a proper solution for evaluating factual accuracy of summaries. Experiments of Zhang et al. (2020b) demonstrate that the attributes of models and datasets can drastically affect the evaluation of factual correctness, and how to design an accurate, model- and data-agnostic evaluation metrics still remains a challenge to the NLP community.

To the best of my knowledge, there have been no studies on the ways that one can secure the completeness of automatically generated text or on how it should be evaluated. However, the big amount of works around text summarization offer a wide range of inspiring ideas and concepts that can lead to the production of a solid tool that will be able to measure the completeness of automatically generated text. Nenkova and Passonneau (2004) proposed Pyramid which was later updated by Yang et al. (2016) who made this method more automatic. These works share the idea that Content Selection is not a deterministic process as different people choose different sentences to include in a summary, and even the same person can select different sentences at different times. Such observations lead to concerns about the advisability of using a single human model and suggest that multiple human gold-standards would provide a better ground for comparison. But for now, human evaluation is the only solution for evaluating completeness, which might be reliable but is also time consuming and not so much practical.

All things considered, a perfect evaluation metric does not exist, and is actually impossible to be built at this moment. However, a metric that takes more aspects of

a summary under consideration can be constructed and might be more useful than *ROUGE*, which only measures the n-gram overlaps between texts.

# Bibliography


Jacopo Amidei, Paul Piwek, and Alistair Willis. Rethinking the agreement in human evaluation tasks. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3318–3329. Association for Computational Linguistics, 2018. URL https://www.aclweb.org/anthology/C18-1281.

Reinald Kim Amplayo, Seonjae Lim, and Seung-won Hwang. Entity commonsense representation for neural abstractive summarization. *CoRR*, abs/1806.05504, 2018. URL http://arxiv.org/abs/1806.05504.

Dzmitry Bahdanau, Kyunghyun Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. *ArXiv*, 1409, 2014.

Regina Barzilay and Kathleen R. McKeown. Sentence fusion for multidocument news summarization. *Comput. Linguist.*, 31(3):297–328, 2005. ISSN 0891-2017.

Anja Belz and Ehud Reiter. Comparing automatic and human evaluation of NLG systems. In *11th Conference of the European Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, 2006. URL https://www.aclweb.org/anthology/E06-1040.

Iram Khurshid Bhat, Mudasir Mohd, and Rana Hashmy. Sumitup: A hybrid single-document text summarizer. In Millie Pant, Kanad Ray, Tarun K. Sharma, Sanyog Rawat, and Anirban Bandyopadhyay, editors, *Soft Computing: Theories and Applications*, pages 619–634. Springer Singapore, 2018. ISBN 978-981-10-5687-1.

Rishi Bommasani and Claire Cardie. Intrinsic evaluation of summarization datasets. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8075–8096. Association for Computational Linguistics, 2020. URL https://www.aclweb.org/anthology/2020.emnlp-main.649.

Jane Bromley, Isabelle Guyon, Yann Lecun, Eduard Scklnger, and Roopak Shah. Signature verification using a "siamese" time delay neural network. 6, 2001.

Ziqiang Cao, Furu Wei, Li Dong, Sujian Li, and M. Zhou. Ranking with recursive neural networks and its application to multi-document summarization. In *AAAI*, 2015.

Ziqiang Cao, Wenjie Li, Sujian Li, Furu Wei, and Yanran Li. Attsum: Joint learning of focusing and summarization with neural attention, 2016.

Chieh-Teng Chang, Chi-Chia Huang, and Jane Yung-jen Hsu. A hybrid word-character model for abstractive summarization. *CoRR*, abs/1802.09968, 2018. URL http://arxiv.org/abs/1802.09968.

Qian Chen, Xiaodan Zhu, Zhenhua Ling, Si Wei, and Hui Jiang. Distraction-based neural networks for modeling documents. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, IJCAI'16, page 2754–2760. AAAI Press, 2016. ISBN 9781577357704.

Yen-Chun Chen and Mohit Bansal. Fast abstractive summarization with reinforce-selected sentence rewriting. *CoRR*, abs/1805.11080, 2018. URL http://arxiv.org/abs/1805.11080.

Jianpeng Cheng and Mirella Lapata. Neural summarization by extracting sentences and words, 2016.

Sumit Chopra, Michael Auli, and Alexander M. Rush. Abstractive sentence summarization with attentive recurrent neural networks. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 93–98. Association for Computational Linguistics, 2016. doi: 10.18653/v1/N16-1012. URL https://www.aclweb.org/anthology/N16-1012.

Junyoung Chung, Çaglar Gülçehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *CoRR*, abs/1412.3555, 2014. URL http://arxiv.org/abs/1412.3555.

Arman Cohan, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Seokhwan Kim, Walter Chang, and Nazli Goharian. A discourse-aware attention model for abstractive summarization of long documents. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 615–621. Association for Computational Linguistics, 2018. doi: 10.18653/v1/N18-2097. URL https://www.aclweb.org/anthology/N18-2097.

Trevor Anthony Cohn and Mirella Lapata. Sentence compression as tree transduction. *CoRR*, abs/1401.5693, 2014. URL http://arxiv.org/abs/1401.5693.

Ferdinand de Saussure. Cours de linguistique générale. *The Modern Language Journal*, 8:317, 1916.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018. URL http://arxiv.org/abs/1810.04805.

Robert L. Donaway, Kevin W. Drummey, and Laura A. Mather. A comparison of rankings produced by summarization evaluation measures. In *Proceedings of the 2000 NAACL-ANLP Workshop on Automatic Summarization*, NAACL-ANLP-AutoSum '00, page 69–78. Association for Computational Linguistics, 2000.

Wafaa S. El-Kassas, Cherif R. Salama, Ahmed A. Rafea, and Hoda K. Mohamed. Automatic text summarization: A comprehensive survey. *Expert Systems with Applications*, 165:113679, 2021. ISSN 0957-4174. doi: https://doi.org/10.1016/j.eswa.2020.113679. URL http://www.sciencedirect.com/science/article/pii/S0957417420305030.

Günes Erkan and Dragomir R. Radev. LexRank: Graph-based lexical centrality as salience in text summarization. *CoRR*, abs/1109.2128, 2011. URL http://arxiv.org/abs/1109.2128.

Angela Fan, David Grangier, and Michael Auli. Controllable abstractive summarization, 2018.

Rafael Ferreira, L. Cabral, R. D. Lins, G. Silva, Fred Freitas, George D. C. Cavalcanti, Rinaldo Lima, S. Simske, and L. Favaro. Assessing sentence scoring techniques for extractive text summarization. *Expert Syst. Appl.*, 40:5755–5764, 2013.

Katja Filippova and Michael Strube. Dependency tree based sentence compression. In *Proceedings of the Fifth International Natural Language Generation Conference*, pages 25–32. Association for Computational Linguistics, 2008. URL https://www.aclweb.org/anthology/W08-1105.

Tao Ge, Furu Wei, and Ming Zhou. Reaching human-level performance in automatic grammatical error correction: An empirical study, 2018.

Jade Goldstein and Jaime Carbonell. Summarization: (1) using mmr for diversity - based reranking and (2) evaluating summaries. In *Proceedings of a Workshop on Held at Baltimore, Maryland: October 13-15, 1998*, TIPSTER '98, page 181–195. Association for Computational Linguistics, 1998. doi: 10.3115/1119089.1119120. URL https://doi.org/10.3115/1119089.1119120.

Charles Greenbacker. Towards a framework for abstractive summarization of multimodal documents. In *Proceedings of the ACL 2011 Student Session*, pages 75–80. Association for Computational Linguistics, 2011. URL https://www.aclweb.org/anthology/P11-3014.

Max Grusky, Mor Naaman, and Yoav Artzi. Newsroom: A dataset of 1.3 million summaries with diverse extractive strategies. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 708–719. Association for Computational Linguistics, 2018. doi: 10.18653/v1/N18-1065. URL https://aclanthology.org/N18-1065.

Karl Moritz Hermann, Tomáš Kočiský, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. Teaching machines to read and comprehend. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'15, page 1693–1701. MIT Press, 2015.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, 1997. ISSN 0899-7667. doi: 10.1162/neco.1997.9.8.1735. URL https://doi.org/10.1162/neco.1997.9.8.1735.

Elad Hoffer and Nir Ailon. Deep metric learning using triplet network. 2014. ISBN 978-3-319-24260-6. doi: 10.1007/978-3-319-24261-3_7.

Wan-Ting Hsu, Chieh-Kai Lin, Ming-Ying Lee, Kerui Min, Jing Tang, and Min Sun. A unified model for extractive and abstractive summarization using inconsistency loss. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 132–141. Association for Computational Linguistics, 2018. doi: 10.18653/v1/P18-1013. URL https://www.aclweb.org/anthology/P18-1013.

Dandan Huang, Leyang Cui, Sen Yang, Guangsheng Bao, Kun Wang, Jun Xie, and Yue Zhang. What have we achieved on text summarization? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 446–469. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.emnlp-main.33. URL https://aclanthology.org/2020.emnlp-main.33.

Karen Spärck Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28:11–21, 1972.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180. Association for Computational Linguistics, 2007. URL https://www.aclweb.org/anthology/P07-2045.

Wojciech Kryscinski, Bryan McCann, Caiming Xiong, and Richard Socher. Evaluating the factual consistency of abstractive text summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9332–9346. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.emnlp-main.750. URL https://aclanthology.org/2020.emnlp-main.750.

Logan Lebanoff, Kaiqiang Song, and Fei Liu. Adapting the neural encoder-decoder framework from single to multi-document summarization. *CoRR*, abs/1808.06218, 2018. URL http://arxiv.org/abs/1808.06218.

L. Lentz and M. De Jong. The evaluation of text quality: expert-focused and reader-focused methods compared. *IEEE Transactions on Professional Communication*, 40(3):224–234, 1997. doi: 10.1109/47.649557.

Jiwei Li, Will Monroe, and Dan Jurafsky. A simple, fast diverse decoding algorithm for neural generation. *CoRR*, abs/1611.08562, 2016a. URL http://arxiv.org/abs/1611.08562.

Wei Li, Lei He, and Hai Zhuge. Abstractive news summarization based on event semantic link network. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 236–246. The COLING 2016 Organizing Committee, 2016b. URL https://www.aclweb.org/anthology/C16-1023.

Chin-Yew Lin. Looking for a few good metrics: Automatic summarization evaluation - how many samples are enough? In *NTCIR*, 2004a.

Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81. Association for Computational Linguistics, 2004b. URL https://www.aclweb.org/anthology/W04-1013.

Hui Lin and Vincent Ng. Abstractive summarization: A survey of the state of the art. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):9815–9822, 2019. doi: 10.1609/aaai.v33i01.33019815. URL https://ojs.aaai.org/index.php/AAAI/article/view/5056.

Peter J. Liu, Mohammad Saleh, Etienne Pot, Ben Goodrich, Ryan Sepassi, Lukasz Kaiser, and Noam Shazeer. Generating wikipedia by summarizing long sequences, 2018.

Yang Liu and Mirella Lapata. Text summarization with pretrained encoders. *CoRR*, abs/1908.08345, 2019. URL http://arxiv.org/abs/1908.08345.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized BERT pretraining approach, 2019.

Elena Lloret, María Teresa Romá-Ferri, and Manuel Palomar. Compendium: A text summarization system for generating abstracts of research papers. *Data & Knowledge Engineering*, 88:164 – 175, 2013. ISSN 0169-023X. doi: https://doi.org/10.1016/j.datak.2013.08.005. URL http://www.sciencedirect.com/science/article/pii/S0169023X13000815.

Elena Lloret, Laura Plaza, and Ahmet Aker. The challenging task of summary evaluation: an overview. *Language Resources and Evaluation*, 52, 2018. doi: 10.1007/s10579-017-9399-2.

Thang Luong, Hieu Pham, and Christopher D. Manning. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421. Association for Computational Linguistics, 2015. doi: 10.18653/v1/D15-1166. URL https://www.aclweb.org/anthology/D15-1166.

Abhishek Mahajani, Vinay Pandya, Isaac Maria, and Deepak Sharma. A comprehensive survey on extractive and abstractive techniques for text summarization. In Yu-Chen Hu, Shailesh Tiwari, Krishn K. Mishra, and Munesh C. Trivedi, editors, *Ambient Communications and Computer Systems*, pages 339–351. Springer Singapore, 2019.

Inderjeet Mani and Mark T. Maybury. *Advances in Automatic Text Summarization*. MIT Press, 1999. ISBN 0262133598.

Kathleen McKeown, Hongyan Jing, R. Barzilay, and Michael Elhadad. Summarization evaluation methods: Experiments and analysis. 1998.

Yashar Mehdad, Giuseppe Carenini, and Raymond T. Ng. Abstractive summarization of spoken and written conversations based on phrasal queries. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1220–1230. Association for Computational Linguistics, 2014. doi: 10.3115/v1/P14-1115. URL https://www.aclweb.org/anthology/P14-1115.

Rada Mihalcea and Paul Tarau. TextRank: Bringing order into text. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 404–411. Association for Computational Linguistics, 2004. URL https://www.aclweb.org/anthology/W04-3252.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed representations of words and phrases and their compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'13, page 3111–3119. Curran Associates Inc., 2013.

N. Moratanch and S. Chitrakala. A survey on extractive text summarization. In *2017 International Conference on Computer, Communication and Signal Processing (ICCCSP)*, pages 1–6, 2017. doi: 10.1109/ICCCSP.2017.7944061.

Martin Mueller. How to write a summary. 2008.

Gabriel Murray, Thomas Kleinbauer, Peter Poller, Tilman Becker, Steve Renals, and Jonathan Kilgour. Extrinsic summarization evaluation: A decision audit task. *ACM Trans. Speech Lang. Process.*, 6(2), 2009. ISSN 1550-4875. doi: 10.1145/1596517.1596518. URL https://doi.org/10.1145/1596517.1596518.

Ramesh Nallapati, Bing Xiang, and Bowen Zhou. Sequence-to-sequence RNNs for text summarization. *CoRR*, abs/1602.06023, 2016a. URL http://arxiv.org/abs/1602.06023.

Ramesh Nallapati, Bowen Zhou, and Mingbo Ma. Classify or select: Neural architectures for extractive document summarization, 2016b.

Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. SummaruNNer: A recurrent neural network based sequence model for extractive summarization of documents, 2017.

Preksha Nema, Mitesh M. Khapra, Anirban Laha, and Balaraman Ravindran. Diversity driven attention model for query-based abstractive summarization. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1063–1072. Association for Computational Linguistics, 2017. doi: 10.18653/v1/P17-1098. URL https://www.aclweb.org/anthology/P17-1098.

Ani Nenkova and Rebecca Passonneau. Evaluating content selection in summarization: The pyramid method. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*, pages 145–152. Association for Computational Linguistics, 2004. URL https://www.aclweb.org/anthology/N04-1019.

Tadashi Nomoto and Yuji Matsumoto. A new approach to unsupervised text summarization. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '01, page 26–34. Association for Computing Machinery, 2001. ISBN 1581133316. doi: 10.1145/383952.383956. URL https://doi.org/10.1145/383952.383956.

Kostiantyn Omelianchuk, Vitaliy Atrasevych, Artem Chernodub, and Oleksandr Skurzhanskyi. GECToR – grammatical error correction: Tag, not rewrite. In *Proceedings of the Fifteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 163–170. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.bea-1.16. URL https://aclanthology.org/2020.bea-1.16.

Tatsuro Oya, Yashar Mehdad, Giuseppe Carenini, and Raymond Ng. A template-based abstractive meeting summarization: Leveraging summary and source text relationships. In *Proceedings of the 8th International Natural Language Generation Conference (INLG)*, pages 45–53. Association for Computational Linguistics, 2014. doi: 10.3115/v1/W14-4407. URL https://www.aclweb.org/anthology/W14-4407.

Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. Technical Report 1999-66, Stanford InfoLab, 1999. URL http://ilpubs.stanford.edu:8090/422/. Previous number = SIDL-WP-1999-0120.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. BLEU: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, page 311–318. Association for Computational Linguistics, 2002. doi: 10.3115/1073083.1073135. URL https://doi.org/10.3115/1073083.1073135.

Annapurna Patil, Shivam Dalmia, Syed Ansari, Tanay Aul, and Varun Bhatnagar. Automatic text summarizer. pages 1530–1534, 2014. doi: 10.1109/ICACCI.2014.6968629.

Romain Paulus, Caiming Xiong, and Richard Socher. A deep reinforced model for abstractive summarization. *CoRR*, abs/1705.04304, 2017. URL http://arxiv.org/abs/1705.04304.

Jeffrey Pennington, Richard Socher, and Christopher Manning. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543. Association for Computational Linguistics, 2014. doi: 10.3115/v1/D14-1162. URL https://www.aclweb.org/anthology/D14-1162.

J. Peter. *Artificial versifying*. 1677. URL https://books.google.gr/books?id=B2Y5yQEACAAJ.

Dragomir R. Radev and Daniel Tam. Summarization evaluation using relative utility. In *Proceedings of the Twelfth International Conference on Information*

*and Knowledge Management*, CIKM '03, page 508–511. Association for Computing Machinery, 2003. ISBN 1581137230. doi: 10.1145/956863.956960. URL https://doi.org/10.1145/956863.956960.

Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using siamese bert-networks. *CoRR*, abs/1908.10084, 2019. URL http://arxiv.org/abs/1908.10084.

Ehud Reiter and Anja Belz. An investigation into the validity of some metrics for automatically evaluating natural language generation systems. *Computational Linguistics*, 35(4):529–558, 2009. doi: 10.1162/coli.2009.35.4.35405. URL https://www.aclweb.org/anthology/J09-4008.

Alexander M. Rush, Sumit Chopra, and Jason Weston. A neural attention model for abstractive sentence summarization. *CoRR*, abs/1509.00685, 2015. URL http://arxiv.org/abs/1509.00685.

Gerard Salton and Christopher Buckley. Term-weighting approaches in automatic text retrieval. *Inf. Process. Manage.*, 24(5):513–523, 1988. ISSN 0306-4573. doi: 10.1016/0306-4573(88)90021-0. URL https://doi.org/10.1016/0306-4573(88)90021-0.

Yogesh Sankarasubramaniam, Krishnan Ramanathan, and Subhankar Ghosh. Text summarization using Wikipedia. *Information Processing & Management*, 50 (3):443 – 461, 2014. ISSN 0306-4573. doi: https://doi.org/10.1016/j.ipm.2014.02.001. URL http://www.sciencedirect.com/science/article/pii/S0306457314000119.

Natalie Schluter. The limits of automatic summarisation according to ROUGE. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 41–45. Association for Computational Linguistics, 2017. URL https://www.aclweb.org/anthology/E17-2007.

Abigail See, Peter J. Liu, and Christopher D. Manning. Get to the point: Summarization with pointer-generator networks. *CoRR*, abs/1704.04368, 2017. URL http://arxiv.org/abs/1704.04368.

Kaiqiang Song, Lin Zhao, and Fei Liu. Structure-infused copy mechanisms for abstractive summarization. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1717–1729. Association for Computational Linguistics, 2018. URL https://www.aclweb.org/anthology/C18-1146.

Nisan Stiennon, Long Ouyang, Jeff Wu, Daniel M. Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul Christiano. Learning to summarize from human feedback, 2020.

Ladda Suanmali, Mohammed Salem Binwahlan, and Naomie Salim. Sentence features fusion for text summarization using fuzzy logic. In *2009 Ninth International Conference on Hybrid Intelligent Systems*, volume 1, pages 142–146, 2009a. doi: 10.1109/HIS.2009.36.

Ladda Suanmali, Naomie Salim, and Mohammed Salem Binwahlan. Fuzzy logic based method for improving text summarization. *CoRR*, abs/0906.4690, 2009b. URL http://arxiv.org/abs/0906.4690.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. Sequence to sequence learning with neural networks. *CoRR*, abs/1409.3215, 2014. URL http://arxiv.org/abs/1409.3215.

Jiwei Tan, Xiaojun Wan, and Jianguo Xiao. Abstractive document summarization with a graph-based attentional neural model. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1171–1181. Association for Computational Linguistics, 2017. doi: 10.18653/v1/P17-1108. URL https://www.aclweb.org/anthology/P17-1108.

Hideki Tanaka, Akinori Kinoshita, Takeshi Kobayakawa, Tadashi Kumano, and Naoto Katoh. Syntax-driven sentence revision for broadcast news summarization. In *Proceedings of the 2009 Workshop on Language Generation and Summarisation (UCNLG+Sum 2009)*, pages 39–47. Association for Computational Linguistics, 2009. URL https://www.aclweb.org/anthology/W09-2808.

Zhaopeng Tu, Zhengdong Lu, Yang Liu, Xiaohua Liu, and Hang Li. Modeling coverage for neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 76–85. Association for Computational Linguistics, 2016. doi: 10.18653/v1/P16-1008. URL https://www.aclweb.org/anthology/P16-1008.

A.M Turing. I.—COMPUTING MACHINERY AND INTELLIGENCE. *Mind*, LIX (236):433–460, 1950. ISSN 0026-4423. doi: 10.1093/mind/LIX.236.433. URL https://doi.org/10.1093/mind/LIX.236.433.

Chris van der Lee, Albert Gatt, Emiel van Miltenburg, Sander Wubben, and Emiel Krahmer. Best practices for the human evaluation of automatically generated text. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 355–368. Association for Computational Linguistics, 2019. doi: 10.18653/v1/W19-8643. URL https://www.aclweb.org/anthology/W19-8643.

Oriol Vinyals, Meire Fortunato, and Navdeep Jaitly. Pointer networks. *ArXiv*, abs/1506.03134, 2015.

Michael Völske, Martin Potthast, Shahbaz Syed, and Benno Stein. TL;DR: Mining Reddit to learn automatic summarization. In *Proceedings of the Workshop on New Frontiers in Summarization*, pages 59–63. Association for Computational Linguistics, 2017. doi: 10.18653/v1/W17-4508. URL https://www.aclweb.org/anthology/W17-4508.

Sharon M. Walter. Review of "evaluating natural language processing systems: An analysis and review" by karen sparck jones and julia r. galliers. springer-verlag 1995. *Comput. Linguist.*, 24(2):336–338, 1998. ISSN 0891-2017.

M. Wang, X. Wang, and Chao Xu. An approach to concept-obtained text summarization. *IEEE International Symposium on Communications and Information Technology, 2005. ISCIT 2005.*, 2:1337–1340, 2005.

S. Wang, X. Zhao, B. Li, B. Ge, and D. Tang. Integrating extractive and abstractive models for long text summarization. In *2017 IEEE International Congress on Big Data (BigData Congress)*, pages 305–312, 2017. doi: 10.1109/BigDataCongress.2017.46.

Qian Yang, Rebecca J. Passonneau, and Gerard de Melo. PEAK: Pyramid evaluation via automated knowledge extraction. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, AAAI'16, page 2673–2679. AAAI Press, 2016.

Fangfang Zhang, Jin-ge Yao, and Rui Yan. On the abstractiveness of neural document summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 785–790. Association for Computational Linguistics, 2018. doi: 10.18653/v1/D18-1089. URL https://www.aclweb.org/anthology/D18-1089.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J. Liu. PEGASUS: Pre-training with extracted gap-sentences for abstractive summarization, 2020a.

Yuhui Zhang, Yuhao Zhang, and Christopher D. Manning. A close examination of factual correctness evaluation in abstractive summarization. 2020b.

Ming Zhong, Pengfei Liu, Yiran Chen, Danqing Wang, Xipeng Qiu, and Xuanjing Huang. Extractive summarization as text matching, 2020.

Qingyu Zhou, Nan Yang, Furu Wei, Shaohan Huang, Ming Zhou, and Tiejun Zhao. Neural document summarization by jointly learning to score and select sentences, 2018.

# A APPENDIX A: Model output Examples

Appendix A provides the reader with some examples of the model output. These examples were chosen in order to show the limitations of the models, used in this research, in practice.

**EXAMPLE 1**

**Article:** The superstitious will know – or perhaps those who have seen Alfred Hitchcock 's classic film the birds – that a bird in the house is a bad omen and one that is said to equate to death .So a family in Texas would be forgiven for being slightly put out about the fact that their home became infested with them .According to the video maker , who returns home with the other people in the clip to find the infestation , the birds came in through the chimney .The video maker stated that he initially thought the intruders were bats but now believes them to be sparrows capturing them on camera as they fly around the room , the man remains very calm while narrating on the carnage before him – birds crashing into the walls and flying overhead .He says : ‘ there are birds everywhere , what the hell , look at this , ’ before reiterating his point and recoiling as one crashes into him .A female voice then speaks out – also extremely calmly considering the fact the front room has been turned into an aviary .Despite the infestation of birds in their front room , the family remain very calm throughout the videoaccording to the video maker the birds managed to invade the house through the chimneythe lady says : ‘ careful not to step on them , ’ as the video maker points out that some of the birds appear to be dying , as they lie on the ground .the filmmaker then states : ‘ this is awful , ’ and suggests that the group need to find a way to get the birds out of the house .The video concludes with the filmmaker continuing to record the birds as they circle the room and bounce off the walls .The video maker states that the scene is awful as a number of birds crash into the walls and end up on the flooraccording to the video maker there were around 30 birds in the front room when the group came home .Speaking on another video , he also states that he initially thought they were bats before realising they were birds and potentially sparrows .

**Text Rank:** According to the video maker , who returns home with the other people in the clip to find the infestation , the birds came in through the chimney .The lady says : ‘ careful not to step on them , ’ as the video maker points out that some of the birds appear to be dying , as they lie on the ground .The video maker states that the scene is awful as a number of birds crash into the walls and end up on the floor according to the video maker there were around 30 birds in the front room when the group came home .

**Fast Abstractive Rewriting:** Superstitious will know those who have seen alfred hitchcock 's classic film .The video maker stated that he thought the intruders were bats but now believes them to be sparrows .The birds came in through the chimney .The man remains very calm while narrating on the carnage .

### Comments

Here one may see that TextRank mentions "the videomaker" and "the lady", while the reader can not be aware of these subjects having not read the actual article. Moreover, the sentences seem to not be connected smoothly with each other. Fast

Abstractive Rewriting's output summary's first sentence is rather weird and irrelevant with the original article.

**EXAMPLE 2**

**Article** : Time fries when you 're having fun .Dedicated waitress Judy Eddingfield , 65 , celebrated her 50th anniversary working at winstead 's restaurant in Kansas city last week and declares that it 's the only job 's she 's ever had or ever wanted .‘ It 's like my home away from home and i just love it here , ' Eddingfield told fox .Dedicated employee : for most of her life Judy Eddingfield has been employed as a server at winstead 's in Kansas city and says it 's the only job she 's ever had or ever wanted a staff member at winstead 's said she believes one of the women in this photograph is Judy when she first started working at the restaurant classic establishment : this is a photo of the the winstead family who own the restaurant showing up for work in 1940 around the time when Judy Eddingfield 's mother started working thereedding field says she remembers when french fries and a classic coca-cola cost just 65 cents at the same restaurant she 's worked at for half-a-century .For Eddingfield , slinging burgers with a smile is a family tradition .On April , 6 , 1965 Eddinngfield began working at the restaurant where she was trained by her mother who had already been employed at the establishment for 13 years .Eddingfield said that her first day on the job when she was just 15-years-old she spilled a shake all over a woman 's fur coat and thought for sure that she 'd be fired .‘ The tray tipped over and vanilla malt slid down a lady 's fur coat .I was so embarrassed , ' Eddingfield told the Kansas city star ‘ of course , she hollered , ' said Eddingfield .All was forgiven after winstead 's , open since 1940 , got the coat cleaned .Eddingfield along with many of her siblings continued to provide top notch service .Loves to serve : Judy Eddingfield says that her kind customers are what make her job worthwhile lots of love : winstead 's honored Judy Eddingfield last week for her wonderful 50 years of working at the classic burger restaurant 'I had two brothers , my sister , my mother , two aunts , two cousins and lots of friends that have worked here , ' said Eddingfield . 'I was three years old when my mother started here , so i 've been eating these burger 60 some years . 'Eddingfield says she even met her first husband outside the restaurant and they married 18 months later .The kansas city star reports that one of Eddingfield 's favorite winstead 's memories was when Jerry Mathers and tony dow of the television series ‘ leave it to beaver ' came to eat there in the late 1970 's .she said that mathers was ‘ talking with his hands ' and knocked sent his food flying everywhere .Marriage : Judy Eddingfield even met her true love who she calls ‘ frog ' at winstead 's and the pair married 18-months-later celebration : dedicated waitress Judy Eddingfield celebrated her 50th anniversary working at winstead 's restaurant in Kansas city last week and declares that it 's the only job 's she 's ever had or ever wanted‘ leave it to beaver , ' his on-screen brother said jokingly , almost as though it were scripted .On a scarier note , Eddingfield says she once was closing up the restaurant when a man started knocking on the door with a gun .She called 911 and hid in the bathroom until the police arrived .Eddingfield 's employees say that as she gets older eddingfield gets more delightful and that she is always willing to share her knowledge of the business .‘ She trained me to be her .Pleasant as she is , prompt .So that people will come here for you and continue to come , ' said server Sue Allen who has been there for 13 years .Close knit team : Eddingfield , pictured with some of her coworkers , has a passion for serving food: winstead 's when it first opened in 1940 has an almost identical style to the winstead 's that 's here today Eddingfield 's biggest tip she ever received was a generous man who treated his entire staff in 2014 and staff agree it was well deserved .Eddigfield turns 66 on May 13 and though she is getting older , she plans to keep on working as long as she is healthy .When asked why she chose to stay at the establishment for so long , she replied that the customers make her job worthwhile .‘ The customers really make it special , ' she says .General manager Kathi Fern says that eddingfield is one of her best employees .‘ It 's just amazing the way she treats her customers , ' said Fern . 'I call her my dinosaur .She 's a dying breed and there will never be servers like her again . ' said fern of her dedicated employee .Classic drive-thru : winstead 's was established in 1940 and to this day is known for serving quality burgers at their drive-thru.

**LEAD3:** Time fries when you 're having fun . Dedicated waitress judy eddingfield , 65 , celebrated her 50th anniversary working at winstead 's restaurant in Kansas city last week and declares that it 's the only job 's she 's ever had or ever wanted . ' it 's like my home away from home and i just love it here , ' Eddingfield told Fox .

**BERTSUM_ABS**: Judy Eddingfield , 65 , celebrated her 50th anniversary working in Kansas city last week .For most of her life Judy Eddingfield has been employed as a server .She remembers when french fries and a classic coca-cola cost just 65 cents .She believes one of the women in this photograph is Judy when she first started working.On April 6 , 1965 Eddinngfield began working at the restaurant where she was trained by her mother .

**Fast Abstractive Rewriting:** Judy Eddingfield , 65 , celebrated her 50th anniversary working in Kansas city last week .For most of her life Judy Eddingfield has been employed as a server .She remembers when french fries and a classic coca-cola cost just 65 cents .She believes one of the women in this photograph is Judy when she first started working.On April 6 , 1965 Eddinngfield began working at the restaurant where she was trained by her mother .

**MatchSum:** Dedicated employee : for most of her life Judy Eddingfield has been employed as a server at winstead 's in Kansas city and says it 's the only job she 's ever had or ever wanted. Eddingfield says she remembers when french fries and a classic coca-cola cost just 65 cents at the same restaurant she 's worked at for half-a-century .

### Comments

This examples shows the incapability of LEAD3 to capture all salient information of a big article. Also, it showcases the confusion of our two main models when it comes to names, as both output summaries state that "Judy Eddingfield , 65 , celebrated her 50th anniversary working in Kansas city..", while they should also include the actual name of the restaurant. MatchSum's output is also included in order to show this model's quality. The first sentence that was chosen is actually the most important sentence of the text, as it includes the main facts that should be mentioned. However, the second sentence contains too specific information about the text, which does not match with the general intentions of a summary.

## EXAMPLE 3

**Article:** Kabul , Afghanistan it is an unimaginably hideous outcome .To be raped by your cousin 's husband ; be jailed for adultery as your attacker was married ; to suffer the ignominy of global uproar about your jailing and assault , but be pardoned by presidential decree ; and then to endure the shame and rejection from a conservative society that somehow held you to blame .The solution in this society ?That 's what happened to Gulnaz , who was barely 16 when she was raped .She 's now carrying the third child of her attacker , Asadullah , who was convicted and jailed – though this was then reduced .Gulnaz 's plight – like so much in beleaguered Afghanistan – disappeared from the world 's gaze once she was pardoned and released courtesy of a presidential pardon .Instead of a new start , what followed for Gulnaz was a quiet , afghan solution to the " problem " – a telling sign of where women 's rights stand in Afghanistan despite the billions that have poured into this country from the u.s. government and its nato allies during more than a decade of war .We found Gulnaz in her family home .Smile , the name of the daughter born of the rape , is now a shining little girl , bouncing around the house that her mother shares with Asadullah 's first wife – who is also Aulnaz 's cousin .Asadullah agreed to let us speak with him and Gulnaz because , it seemed , he wanted to show us that things were now settled , that under Afghanistan 's version of social morality he had done the right thing .He had rescued Gulnaz from shame ." If i had n't married her , ( but ) according to our traditions , she could n't have lived back in society , " he tells us ." Her brothers did n't want to accept her back .Now , she does n't have any of those problems . "2011 : thousands sign petition for gulaz release Gulnaz remains subdued throughout our meeting and does not once look her husband in the eye ." I did n't want to ruin the life of my daughter or leave myself helpless so i agreed to marry him , " she says ." We are traditional people .When we get a bad name , we prefer death to living with that name in society . "As Smile attempts to pour tea , the other seven children in this household run around the courtyard .The first wife remains unseen in the house .A portrait of Gulnaz 's liberator in 2011 , the then-president Hamid Karzai , hangs on the wall .But the sense of order here is undermined by the fact that this is a house built around a crime .How gulnaz ended up here requires some explanation .There was pressure upon her to marry her attacker after her release .But at the same time , other activists were trying to assist her with an asylum bid abroad ." Unfortunately , Gulnaz was heavily pressured to marry her attacker by various people within the government which , in and of itself , was immensely disappointing , " her former attorney , an american citizen named kimberley motley , tells us ." Gulnaz was constantly told that neither she nor her daughter would be protected if she did not succumb to their pressure to marry ... Gulnaz essentially became a prisoner of her environment ." As an uneducated , young , single mother with no family support , it would have been an uphill battle for gulnaz and her daughter . "She was introduced to her attacker in the shelter where cnn first interviewed her upon release from prison .They talked and it was agreed she would marry him .Most disturbingly , the woman who – despite knowing the stigma it would create around her – defiantly insisted she had been raped when we spoke nearly four years ago , now says she was told by her relatives to make up the allegations ." Now she is beside me and knows that it was not as big as they had shown it , " says Asadullah ." No I am not thinking about it anymore , " Gulnaz adds ." I do n't have a problem with him now and i do n't want to think about the past problems .My life is ok ... i am happy with my life ... it is going on . "She is then permitted to talk with us alone .Asadullah moves away but stands close to the door of the room .Though she now maintains she was not raped , she explains her decision .she contradicts her husband , saying her brothers would have taken her back , had she not married him ." My brothers opposed the marriage and told me to take my daughter and go to Pakistan to live with them instead , " she says ." But now we 're married , they disowned me and wo n't see me again . "Her decision was for her daughter ." No , i could n't fulfill my wishes in life .I married this man ; I cut relations with my family only to buy my daughter 's future . "It is truly chilling to see how things have gone for Gulnaz after the level of international attention her story received – pregnant with the third child of the man who was once her rapist , accepting a life as his second wife , trapped in his home .

**Fast Abstractive Rewriting:** To : to suffer the ignominy of global uproar about your jailing and assault . Gulnaz 's plight is like so much in beleaguered Afghanistan , she says . She 's now carrying the third child of her attacker , who was convicted . She is now a shining little girl , bouncing around the house that her mother shares with asadullah 's first wife .

**Reference:** Gulnaz was jailed after the attack as her rapist was married. Her case gained international attention ; prompted a presidential pardon she was forced to marry her attacker or face disgrace.

**BERTSUM_ABS:** Gulnaz was raped by her cousin 's husband and jailed for adultery. Her plight disappeared from the world 's gaze when she was pardoned. A quiet , afghan solution to the p̈roblem ïs a sign of where women 's rights stand in Afghanistan.

### Comments

As one may observe, Fast Abstractive Rewriting model's output summary contains a strange "To : to.." phrase, while the second sentence is ungrammatical and irrelevant as well. Also, we can see that the sentences are not at all connected with each other. On the other hand, the reference summary is very clear and accurate. However,

as mentioned in Section 5.2, it is not actually a summary, but a set of connected highlight sentences of the text. BERTSUM_ABS is included in this example in order to show a good instance of this algorithm's outputs.

**EXAMPLE 4**

**Article:** Jordan Henderson is not ready to give up on Liverpool 's champions league hopes despite an eight-point gap to Manchester united in fourth place . The 24-year-old scored a penalty at the emirates on Saturday , but it was not enough as liverpool fell to a 4-1 thrashing at the hands of arsenal to leave their top-four hopes in tatters . That result followed a devastating 2-1 home defeat by Manchester United last time out and Brendan Rodgers ' side have fallen from two points off champions league qualification to eight behind . Jordan Henderson urged liverpool not to give up on their hopes of finishing in the top four Henderson is targeting seven wins out of seven in the league and an immediate response at blackburn but Henderson urged his team not to give up on the top-four race with seven games to play , and to bounce back immediately on wednesday night when they take on blackburn in an fa cup quarter-final replay . He told liverpool 's official website : ' We knew it was going to be difficult [ to finish in the top four ] , even before saturday , but it makes it even more difficult . ' That does n't mean we 'll just give up . We 've got seven games left now and we need to try to win every one . If we do that , then you never know . ' Blackburn is another big game for us straight away . We 've got to recover well , learn from this one and move on and try to get a big win on wednesday . '

**Fast Abstractive Rewriting:** Jordan Henderson is not ready for liverpool 's champions league . Liverpool beat arsenal 4-1 at the emirates on saturday . Henderson is targeting seven wins out of seven . Jordan Henderson urged liverpool to give up on their hopes of finishing in the top four .

**Comments**

In this example, the output summary of Fast Abstractive Rewriting is not good since it starts by giving wrong information to the reader. "Jordan Henderson is not ready for liverpool's champions league." and "Jordan Henderson urged liverpool to give upon their hopes of finishing in the top four." are wrong sentences as in the original article Jordan Henderson is actually not ready to give up on Liverpool's Champions League hopes. Also the first of the two sentences does not make any sense semantically.

# B APPENDIX B: Novel words

Appendix B provides the reader with two wordclouds that represent the novel words (do not exist neither in the article nor in the reference summary) of Fast Abstractive Rewriting (Figure B.1) and BERTSUM_ABS (Figure B.2).

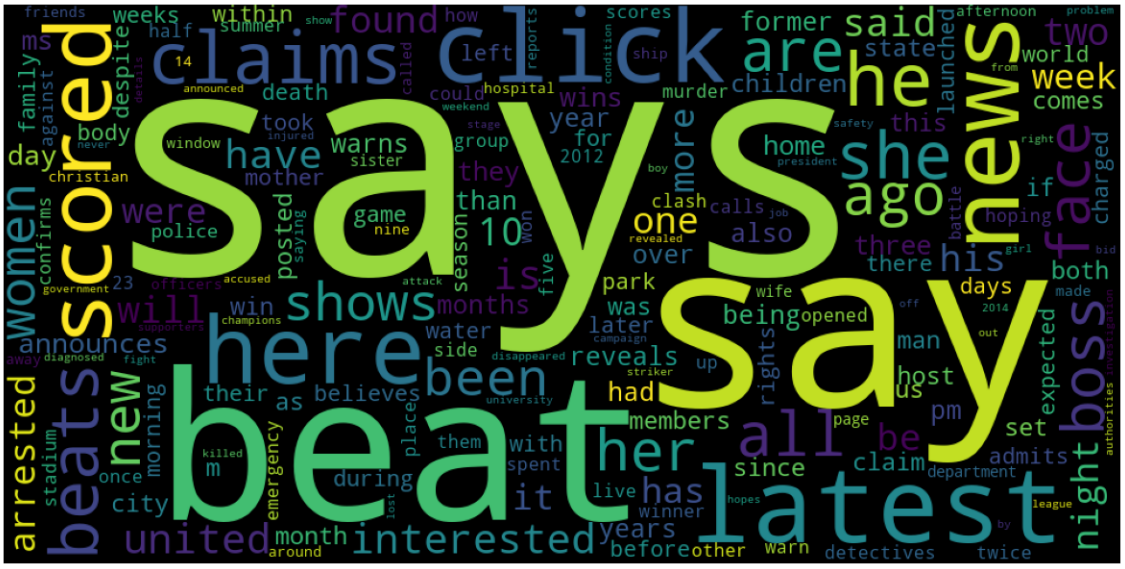


Figure B.1: Wordcloud of the novel words that were generated by Fast Abstractive Rewriting model.

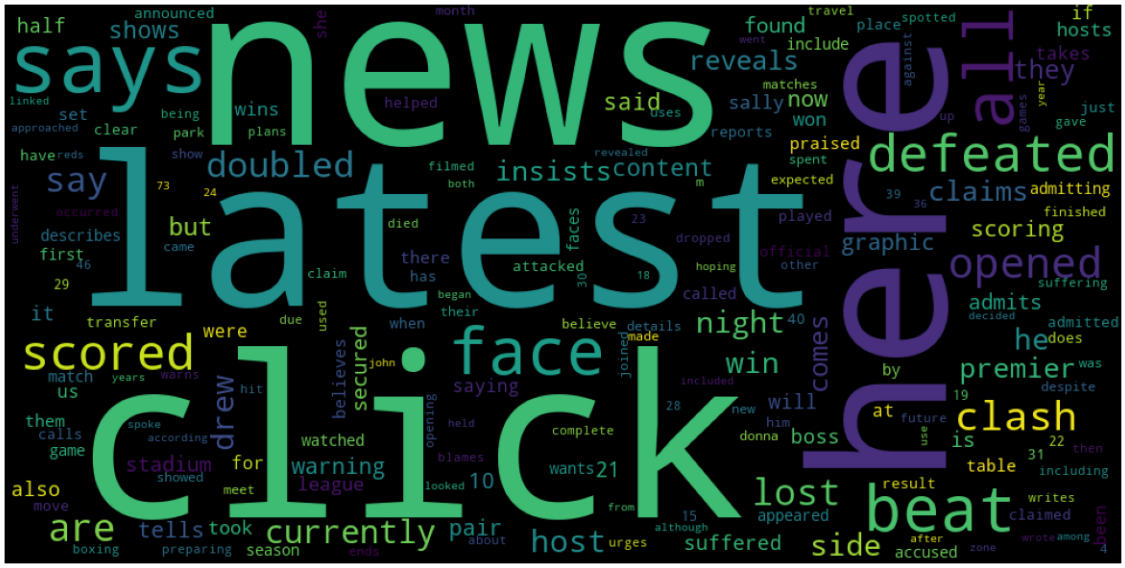


Figure B.2: Wordcloud of the novel words that were generated by BERTSUM_ABS.

# C APPENDIX C: Properties of the participants

| ID | Age | Education | Reading Frequency (1-5) | Nationality | English Level |
|---|---|---|---|---|---|
| User 1 | 45+ | PhD | 4 | Greece | C2 Proficient |
| User 2 | 24-35 | MSc | 4 | Italy | C1 Advanced |
| User 3 | 24-35 | BSc | 2 | Greece | B1 Intermediate |
| User 4 | 18-24 | MSc | 5 | Greece | C2 Proficient |
| User 5 | 24-35 | MSc | 3 | Greece | C2 Proficient |
| User 6 | 24-35 | MSc | 4 | Greece | C2 Proficient |
| User 7 | 24-35 | MSc | 5 | Greece | C2 Proficient |
| User 8 | 24-35 | BSc | 2 | Greece | B2 Upper Intermediate |
| User 9 | 24-35 | MSc | 4 | The Netherlands | C2 Proficient |
| User 10 | 24-35 | MSc | 4 | Greece | C2 Proficient |

Table C.1: Properties of the Human Evaluation participants. "Reading Frequency" represents the frequency that participants read English news articles.