Adapting THExt to news and articles: several approaches to tackle redundancy

Domenico Bulfamante
Politecnico di Torino
Student ID: s301780
domenico.bulfamante@studenti.polito.it

Giovanni Sciortino
Politecnico di Torino
Student ID: s302959
giovanni.sciortino@studenti.polito.it

Giuseppe Suriano
Politecnico di Torino
Student ID: s296605
s296605@studenti.polito.it

***Abstract*—Text Highlighting involves selecting certain important sentences from a document that help provide a good overview of the entire document. With these experiments we tried to implement modules that increase the performances of the baseline model (THExt fineturned on CNN/Daily Mail). In fact we introduced an efficient context extractor, different redundancy algorithms and an alternative training process that uses BERT as a ”static” encoder. Considering all these experiments we achieved a 31,1% improvement on the evaluation task with respect to the baseline model.**

The code of the work can be found in the following link here

I. Introduction

Text highlights are short sentences that are able to summarize the whole text, encapsulating the most salient part or significant insights. In the last years, the task of highlights extraction became more and more relevant, for example, the extraction of paper highlights for journal articles or scientific papers. To do so, we decide to propose some extension of the THExt model proposed in [1]. This model consists in an extractive sentence-based summarization strategy whose goal is to select 3 to 5 existing sentences that can be recommended as candidate highlights. More in particular this structure is fine-tuned on scientific paper highlights extraction, which is based on an established contextualized embedding architecture, namely BERT [2].

BERT generate sentence-level text encoding by leveraging the *attention mechanism*, which enables the sequence encoder to attend to specific portions of the text sequence while processing a specific word. Moreover, the highlights extraction is implemented as a regression task, whose purpose is to find the candidate sentences that maximize their expected similarity with the humanly generated annotations.

To give a brief overview of our work: first, we performed some experiments with the CNN/Daily Mail dataset, which has been used for evaluating summarization on various models as a reference benchmark. The dataset contains online news articles paired with multi-sentence summaries. So the aim is to understand how good the model is on articles highlighting.

Then, we propose several methods to tackle the redundancy over the extracted sentences thanks to the usage of BERTScore [3], which can better capture the semantic meaning of a sentence compared to standard syntactical metrics that rely on n-grams overlapping, like Rouge [4].

Another implementation was on the definition of the context for our model. In the [1] an attempt was done; in that case the best context was the abstract of the specific paper. We wanted to go deeper and try in a different way to feed the whole text as context for our candidate sentence. For this purpose BART [5] have been chosen. Thanks to the very good performance in summarization we decide to use the summary generated by BART as context, since it represents better article-level information useful for the model to properly evaluate each candidate sentence.

II. Related Works

A. Summarization

Text summarization is the task of generating a short length and fluent summary of a longer piece of text. Neural networks have been used to perform this task and achieved great success. There are two main approach: *abstractive* that provides a summary from scratch or *extractive* that selects the key phrases from the former text. Clearly, the purpose of both approaches is to deliver an accurate summary that captures the key points and subject of the original text, while being more concise than the initial text and easier to read.

Recent research work on extractive summarization spans a large range of approaches. Some of them instantiate their encoder-decoder architecture by choosing RNN (Nallapati et al., 2017 [10]; Zhu et al., 2018 [11]) or Transformers. In fact, in 2019, Liu and Lapata [12]proposed a method for fine-tuning large, pre-trained transformer models on a small dataset of labeled summaries can lead to significant improvements in performance. The success of architectures are mainly referable to the mechanism that transformers relies on, i.e. *Attention mechanisms*. In particular, the latter allows to model dependencies without taking care about their distance in input or output sequences.

Similarly, by relying on Transformers, in 2022 Cagliero and La Quatra proposed THExt [1], a structure that leverages the attention mechanism adopted in transformer models to improve the accuracy of sentence relevance estimation. Unlike existing approaches, it relies on training of a deep regression model. To attend patterns relevant to highlights content it also enriches sentence encodings with a section-level contextualization since the architecture was tailored essentially for scientific papers.

B. Transformer Based Summarization

Pre-trained language models such as BART [5] often produce really great results on the summarization tasks. In fact, this approach has gained widespread popularity in recent years due to the success of several state-of-the-art models. However, they are used on short news articles such as XSum [13] or CNN/Daily Mail [14] datasets. These models are not designed for scientific articles and their space/computational complexity grows in a quadratic way with the size of the input.

Other extractive summarizer model like HIBERT [15] gained great success by basically learning about context aware sentence representations using multiple layers of transformers. Here, 15% of the sentences in the document are randomly selected and masked (replaced with a single [mask] token) with the goal to predict the sentence embedding of the masked sentences. Moreover, BERTSUM [18] is another BERT style extractive summarizer that extends BERT to multiple sentences by expanding the positional embedding and using interval segmentation embeddings to distinguish multiple sentences within a document.

In 2020 Sotudeh et al. [16], in their paper about LongSumm, added section information to the objective function of BERTSUM so it could optimize both the sentence prediction and section prediction tasks in a multi-task setting. However, most of these transformer-based extractive summarizers do not scale for long documents with thousands of tokens nor can they be applied to many full scientific documents.

III. Methods

Our proposed model relies on the architecture of THExt that leverages contextualized embeddings and transformer models in the extraction of relevant paper highlights. In the specific, it is built on BERT that makes use of a bidirectional encoder representation, which relies on the attention mechanism to learn contextual relationships between words.

More in particular, the structure of THExt consists of the following parts:

- Context definition
- Sentence encoding based on BERT
- Regression based on Fully Connected Neural Network (FCNN)

The context definition is a crucial part of the architecture. This is due to the fact that the context is useful to the model to encapsulate global context dependencies in order to properly evaluate the candidate sentences. Hence, the considered snippet of text is expected to include a discussion on the main concepts of a document.

Since THExt is an architecture designed specifically for highlights extraction of scientific paper, for this task the best candidate to be selected as context is the abstract of a single paper, as shown in the results of [1]. In our case, we are deploying this architecture on CNN/Daily Mail (will further discuss in section 4); the documents in this dataset are news articles, that are less structured than a scientific paper and do not contain an abstract-like section. In order to tackle this problem and define a proper context, we propose a feasible solution that relies on BART. Further details about this solution will be given in the following sections (IV-A).

Sentence encoding

Once the context is defined for a document, we deploy BERT to encode each sentence in the input text in the following manner: we concatenate the candidate sentence s^* with the context previously defined $\mathcal{C}$ separated by a [SEP] token. The main idea is to capture not only underlying local pattern in s^*, but also the global context dependencies summarized by $\mathcal{C}$. To this purpose, thanks to the attention mechanism , the model conveniently exploits the additional context description to attend relevant token-sequences of the candidate sentence thus focusing model learning on the most discriminating patterns.

Given the linear projections Q, K, V, the Transformer [17] computes the attention scores as follows:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)$$

The attention scores are computed by considering both the local candidate sentence and the paper context.

Regression

Once BERT produce the encoded sentences, we feed the embedding into a Fully Connected Neural Network , which is used to train the regression model. The goal is to learn a scoring function from the training set that can be used to extract an output summary maximizing the similarity between the selected sentences and the expected highlights. The regression model is trained to minimize the mean square error between the predicted and expected similarity scores. In the specific we quantify the regression function as the syntactical Rouge similarity score between the contents of the candidate sentence and the article highlights. Once the relevance score for each candidate sentence is computed, it is exploited to drive the highlights extraction.

Redundancy approach

The challenge of minimizing redundancy is common in summarization tasks. In THExt, since the top n sentences are selected, it could happen that sentences with higher scores might be very similar, both semantically and syntactically. On the other hand, highlights should be representative sentences but expressing different concepts. This has led us towards the research for an effective method that is capable of identifying redundancy among the extracted sentences and selecting the most important ones, always respecting the definition of highlights described earlier.

We propose three methods to tackle the redundancy in the highlights extracted by the model described previously:

- *iterative method*: in this approach, starting from the higher ranked highlights regarding the predicted sentence scores, we compute iteratively the BERTScore between

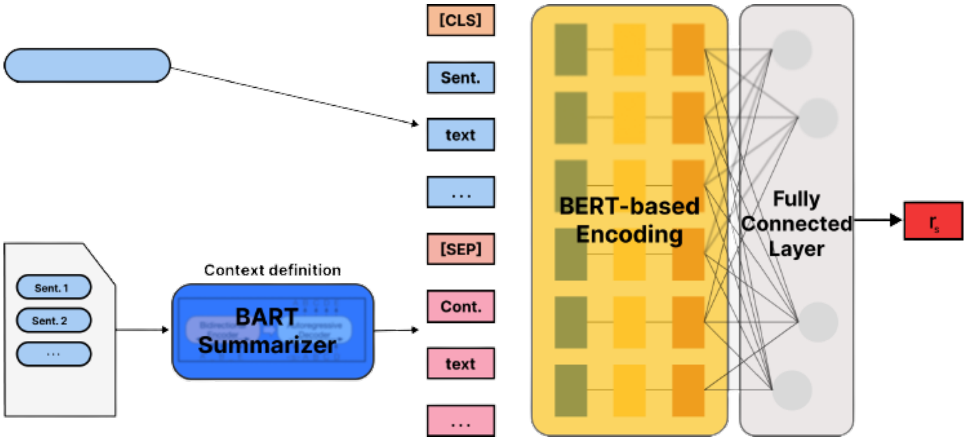

Fig. 1. The presented architecture relying on BERT as sentence encoder and BART as context extractor

the already taken highlights and the remaining ranked sentences; then we pick the higher ranked candidate excluding the one with the higher BERTScore.

- *cluster-based*: this method is based on the concept of clustering, so that similar sentence will be in the same cluster. So once the similarity matrix is computed among the all the sentences with BERTScore, with an hierarchical clustering method we get k number of clusters as the number of highlights we want to extract. Then we select from each cluster the sentence with the highest predicted sentence scores to compose the final highlights.
- *trigram blocking*: Trigram Blocking is widely used in extractive summarization models on the news dataset [18]. In particular, given the predicted sentence scores, instead of just selecting sentences greedily according to the scores, the current candidate is added to the summary only if it does not have trigram overlap with the previous selected sentences. Otherwise, the current candidate sentence is ignored and the next one is checked, until the length limit is reached.

The metric used for evaluate similarity in both iterative and cluster-based methods is BERTScore [3], which can better capture the semantic meaning of a sentence compared to standard syntactical metrics that rely on n-grams overlapping, like Rouge [4].

IV. Experiments

A. CNN/Daily Mail

The CNN/Daily Mail dataset contains human generated abstractive summary bullets generated from news stories in CNN and Daily Mail websites. Three versions are available and we used the not anonymized one. The full datasets has 286,817 training pairs, 13,365 validation pairs and 11,490 test pairs. However, due to the limited computational resources, we didn't considered the training set but we rather considered the whole test set and a subset of 1,600 articles of the validation set as test set.

For the transfer learning task we preprocessed the testing split of the original dataset in this way (around 10,000 documents are taken): sentence tokenizing, context extraction (considering the first sentences that fits the BERT input size), six rouge scores (Rouge-2 and Rouge-l with precision, recall and f1).

For the abstractive context fine-tuning we follow the same procedure as the previously described with a significant difference on the context; in fact in this case the context of each article is generated by BART.

For the testing of the performance of each model we used the around 1,600 articles of the validation split of the original dataset and we replicated the same procedure but replacing the rouge scores with the corresponding gold summary.

B. Experiment setup

First we performed a transfer learning task by allowing THExt to extract highlights on a different dataset. For each sentence we considered as context the first part of the specific article until we reach the maximum input size of BERT concatenating it with the specific sentence.

Moreover, considering BART as context extractor we generate an abstractive summary handling the problem of article's lenght by splitting the article according to BART's maximum input size and his tokenizer; finally for a specific article's summary we concatenate the several summaries given by BART and compose a final one used as context.

These two implementation are both fine-tuned the whole test set of the original dataset of CNN/Daily Mail.

Finally we further perform a fine-tuning on THExt that involves only the weights of the FCNN that will be later motivated in the sect V-B by simply imposing to not update the gradients of the BERT weights. In this case when we backpropagate the loss from the FCNN the resulting gradient on the BERT weights is null.

V. Conclusion

A. Results

All the test made regarding the results that will be discussed in this section is performed on a subset of the validation set of the original CNN/Daily Mail dataset.

TABLE I
COMPARISON OF RESULTS

Methods	Rouge-1 f1 score (%)	Rouge-2 f1 score (%)	Rouge-l f1 score (%)
Model trained updating BERT's weights			
THExt default	22.45	5.93	18.18
THExt+BART default	25.44	7.72	20.80
Model trained with BERT as encoder			
THExt default	23.87	6.91	19.43
THExt clustering	23.70	6.52	19.22
THExt iterative	24.16	7.05	19.60
THExt trigram blocking	23.92	6.82	19.35
THExt+BART default	**29.43**	**11.38**	**24.37**
THExt+BART clustering	27.77	9.60	22.86
THExt+BART iterative	29.30	11.22	24.13
THExt+BART trigram blocking	29.26	10.97	23.98
Oracle	*33.83*	*14.38*	*27.92*

Each model considered in the evaluation phase extracts three sentences as components of the summary. In order to evaluate a model we decide to compute the average Rouge scores between the concatenation of the sentences selected and the gold summary.

From the Table I we can notice that the architecture of THExt have an overall boost in performance on all the KPIs we took in consideration(Rouge-1, Rouge-2 and Rouge-l). Moreover we can see that the use of BART as context extractor the architecture is fundamental to reach comparable results with respect to the Oracle highlights.

For what concerns the methods to tackle redundacy in the highlights extracted we can see that the three approaches implemented achieve comparable results to the default method (first three sentences with the highest predicted sentence scores). A noticeable increment in performance can be seen using these methods when we doesn't use BART as context extractor. Overall between the three methods, the iterative and trigram blocking ones performs better than the clustering.

B. Discussion

The initial transfer learning task was helpful to set the baseline for the whole experiments carried out in this work. The deployment of BART as context extractor was meaningful for the model as discussed in section V-A, this behaviour is justified by the fact that BART is one of the best abstractive text summarizer; in fact in the design phase of this work, we thought about several approaches to give a more appropriate context to THExt discarding the extractive approaches and considering the abstractive one to not bias THExt (using extractive context could introduce bias for the sentences present in the context).

For what concerning the implementations to tackle the redundancy in the highlights extracted we decided to pursuit implementations that doesn't consider strictly the scores given by a semantic metric; but rather working on the predicted sentences scores given by THExt. As explained in the section IV-B in all the approaches considered we pick by default the higher ranked sentence regarding the predicted scores; this fact is a strong assumption that may lead to lower performance since we try to extract 3 highlights (discussed in IV-B).

The worst performance was given by the clustering implementation since news article are typically short, with few sentences, so the clustering approach for these kind of document is not suitable.

Trigram blocking approach as discussed in III is a more heuristic approach that relies on the assumption that the sentences composing the gold summary does not present overlapping trigrams among them. Basically this approach try to solve a semantic problem exploiting syntactic level information. We can conclude that this approach is not that suitable to tackling the redundancy issue despite the fact that the performances using this approach are comparable to the iterative one as discussed before.

For the last approach, iterative, since the condition to discard a candidate is particularly strict so in practice this method leads to extract highlights in the highest ranked candidates. Moreover we notice that this method can be considered as the final approach for redundancy because it is the best performing approach without considering BART as context extractor.

Moreover we can make some considerations about the training approaches we used. In fact initially in the fine-tuning phase, all the weights of the architecture were updated (in the specific case we backpropagate the loss to update the weights of BERT and the FCNN for the regression). So we are considering an other approach that use BERT in the THExt model as a sentence encoder, in the fine-tuning phase just the FCNN weights were updated.

In the first case the embedding of a sentence changed during the fine-tuning process and reasonably the FCNN has to adjust its weights to decrease the loss. In fact considering the fine-tuning losses of each epoch for both approaches we can notice that in the initial approach the losses does not follow a decreasing trend due to the change of the embedding of the same sentence; instead in the second approach the losses for each epoch follow a decreasing trend since the embeddings of the same input remain the same.

Finally we can make some consideration about why the model doesn't reach state of the art performances in extractive summarization for CNN/Daily Mail dataset; in fact the process of highlight extraction is based on the assumption of THExt architecture that considers the best candidates the ones with the highest predicted sentence score to compose the final highlights. This assumption is not consistent if the gold summary is not an extractive one. (composed of extracted sentences)

Overall the performances of our transfer learning implementation using BART as context generator reaches performance near to the oracle ones, giving evidence of a proper and effective implementation.

References

[1] L. Cagliero, M. La Quatra, Transformer-based highlights extraction from scientific papers.

[2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova, BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.

[3] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, Yoav Artzi, BERTScore: evaluating text generation with bert.

[4] Chin-Yew Lin, ROUGE: A Package for Automatic Evaluation of Summaries.

[5] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, Luke Zettlemoyer, BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension.

[6] Rush et al. in 2015, A Neural Attention Model for Abstractive Sentence Summarization.

[7] Abigail See, Peter J. Liu, Christopher D. Manning in 2017, Get To The Point: Summarization with Pointer-Generator Networks

[8] Eva Sharma, Chen Li, Lu Wang in 2018, BIGPATENT: A Large-Scale Dataset for Abstractive and Coherent Summarization

[9] Jianpeng Cheng, Mirella Lapata in 2016, Neural Summarization by Extracting Sentences and Words

[10] Ramesh Nallapati, Feifei Zhai, Bowen Zhou in 2016, SummaRuNNer: A Recurrent Neural Network based Sequence Model for Extractive Summarization of Documents

[11] Haoran Li, Junnan Zhu, Jiajun Zhang, Chengqing Zong, in 2018, Ensure the Correctness of the Summary: Incorporate Entailment Knowledge into Abstractive Sentence Summarization

[12] Yang Liu, Mirella Lapata in 2019, Text Summarization with Pretrained Encoders

[13] Shashi Narayan Shay B. Cohen Mirella Lapata, in 2018, Don't Give Me the Details, Just the Summary! Topic-Aware Convolutional Neural Networks for Extreme Summarization

[14] Danqi Chen and Jason Bolton and Christopher D. Manning in 2016, A Thorough Examination of the CNN/Daily Mail Reading Comprehension Task

[15] Xingxing Zhang, Furu Wei and Ming Zhou in 2019, HIBERT: Document Level Pre-training of Hierarchical Bidirectional Transformers for Document Summarization

[16] Sajad Sotudeh, Arman Cohan, Nazli Goharian in 2020, On Generating Extended Summaries of Long Documents

[17] Ashish Vaswani, et al, Attention Is All You Need.

[18] Yang Liu in 2019, Fine-tune BERT for Extractive Summarization.