Transformer-based highlights extraction from scientific papers

Transformer-based highlights extraction from scientific papers

Moreno La Quatra, Luca Cagliero*

Dipartimento di Automatica e Informatica, Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129, Torino, Italy

Abstract

Highlights are short sentences used to annotate scientific papers. They complement the abstract content by conveying the main result findings. To automate the process of paper annotation, highlights extraction aims at extracting from 3 to 5 paper sentences via supervised learning. Existing approaches rely on ad hoc linguistic features, which depend on the analyzed context, and apply recurrent neural networks, which are not effective in learning long-range text dependencies.

This paper leverages the attention mechanism adopted in transformer models to improve the accuracy of sentence relevance estimation. Unlike existing approaches, it relies on the end-to-end training of a deep regression model. To attend patterns relevant to highlights content it also enriches sentence encodings with a section-level contextualization. The experimental results, achieved on three different benchmark datasets, show that the designed architecture is able to achieve significant performance improvements compared to the state-of-the-art.

Keywords: Highlights Extraction, Transformer model, Extractive Summarization

*Corresponding author. Email: luca.cagliero@polito.it. Phone number: +39 011 090 7179. Fax: +39 011 090 7099

Email addresses: moreno.laquatra@polito.it (Moreno La Quatra), luca.cagliero@polito.it (Luca Cagliero)

1. Introduction

Highlights are 3 to 5 short sentences that convey the core findings of the research presented in a scientific paper. Compared to the abstract of the paper, highlights differ in (1) *structure*: they consist of an ordered list of short sentences (maximum 85 characters per highlight) instead of a narrative section; (2) *goal*: they provide result-oriented insights rather than a comprehensive paper overview; (3) *sentence-level content dependencies*: sentences are separated and, generally, there are no content-level dependencies among them [1].

To help increase the discoverability of journal articles via search engines some journal editors have recently made the submission of paper highlights compulsory (see, for example, [2]). However, the largest amount of past scientific papers are still not annotated. With the twofold aim at enriching existing papers with additional metadata and supporting the manual annotation of newly published works, in this paper we propose an automated approach to extract highlights from the paper full-text. To this end, we design an *extractive* sentence-based summarization strategy whose goal is to select 3 to 5 existing paper sentences that can be recommended as candidate highlights. As discussed later on, the problem of automatically extracting paper highlights is relatively new and leaves room for several research contributions.

The peculiar characteristics of scientific paper highlights call for new Deep Natural Language Processing methods that are aimed at analyzing the underlying dependencies between the sentence-level paper content and the manually annotated highlights. To achieve this goal, we model highlights extraction as a regression task, whose purpose is to find the candidate sentences that maximize their expected similarity with the humanly generated annotations. This work specifically addresses highlights extraction from English-written papers, as English is known to be the standard language of science.

Challenges. Prior work extracts paper highlights by adopting recurrent LSTM-based models trained on word-level encodings of ad hoc linguistic features [1, 3]. However, defining the right context of words and understanding the underlying sentence meanings can be challenging because of the following reasons [4]: (1) Recurrent models tend to neglect long-range text dependencies [5]. (2) The feature engineering steps are strongly related to the subject and thus not easily portable to other research areas.

Method. This paper presents a *Transformer-based Highlights Extractor* (THExt, in short). It is a new approach to extracting highlights based on an established contextualized embedding architecture, namely the transformers [6]. Transformers generate sentence-level text encodings by leveraging the attention mechanism, which enables the sequence encoder to attend specific portions of the text sequence while processing a specific word [5]. Specifically, the pre-training phase relies on a latent text representation inferred directly from the raw input sequence, i.e., without the need to extract handcrafted features. Next, a fine-tuning step, applied on top to the pre-trained model, also considers long-range dependencies between the sentence content and the prediction target that are potentially neglected by recurrent models.

The sentence-level text managed by transformer encoders like BERT [6] has a maximum number of input tokens (typically, 512). This limits the scope of the attention mechanism to bounded text snippets, often located within the same section. Neglecting the global, paper-level sentence contextualization can be harmful in highlights extraction because the text dependencies extracted from multiple sentences and sections are useful for rewarding specific sentence-level patterns. Thus, we propose to enrich the sentence-level encoding with contextual knowledge extracted from different sections. By leveraging paper sectioning we automatically extract the snippets of paper full-text that are most likely to cover highlight-related content.

Results. We conducted an extensive empirical assessment of the performance of the proposed approach on three benchmark datasets (i.e., CSPubSumm [1], BIOPubSumm and AIPubSumm [3]). THExt performs significantly better than state-of-the-art highlights extractors. For example, against the best performer +0.056 (+17%) on CSPubSumm, +0.028 (+10%) on BIOPubSumm, +0.035 (+12%) on AIPubSumm in terms of ROUGE-L F1-score.

Main contributions.

- We propose a new approach, based on Transformer-based encoding, to highlight extraction. To the best of our knowledge, this is the first attempt to use transformer architectures to address automatic highlight generation.
- We design a context-aware sentence-level regressor, in which the semantic similarity between candidate sentences and highlights is estimated by also attending the contextual knowledge provided by the other paper sections.

- We achieve performance superior to state-of-the-art highlights extraction methods on three benchmark datasets. The THExt source code and a set of highlight examples generated from real journal and conference papers are freely available, for research purposes[1].

Toy example. We used THExt to extract the highlights of the present paper. For example, the following top-2 sentences have been selected: *We propose a novel Transformer-based Highlights Extractor (THExt, in short)* and *We achieve performance superior to state-of-the-art highlights extraction methods on three benchmark datasets.* Paper authors can slightly trim or adjust the shortlisted original paper sentences to meet the character length constraint (e.g., the second sentence can be trimmed as follows: *We achieve performance superior to state-of-the-art highlights extraction methods*).

2. Related work

Highlights extraction. The problem of automatically extracting paper highlights from the raw paper content is relatively new. To the best of our knowledge, the first attempt to extract highlights from a scientific paper has been made by [1]. It performs binary classification based on a combination of LSTM-based embeddings and a customized set of summarizing linguistic features. Specifically, the classifier predicts whether a sentence can be recommended as paper highlight or not. To generate the training data it picks the 20 sentences per paper that are most similar to the manually generated paper highlights, in terms of ROUGE-L similarity [7]. The main drawbacks of the aforesaid method are enumerated below.

1. Since the problem is formulated as a binary classification task, the proposed approach does not rank the output sentences.
2. The LSTM-based sentence encoding is not able to attend long-range text dependencies.
3. The abstract content is encoded using word-level embeddings, which are static and independent of the context in which each word appears.

To tackle the issue (1), [3] propose to train a regression model per paper. Each regressor predicts the Rouge-based syntactical similarity between each paper sentence and the corresponding highlights. Then, the most similar

[1] https://github.com/MorenoLaQuatra/THExt (latest access: May 2022)

sentences per paper are returned. The regression models are trained on a set of syntactical and semantic features derived from word-level embeddings. Notice that in [3] the authors first used pre-trained BERT-based models to generate the sentence representations and then run unsupervised algorithms on top of them. Hence, issues (2) and (3) are still open. Conversely, in the present work, we propose a supervised approach, based on end-to-end regressor training. The model estimates, for each sentence in the paper, the corresponding relevance score. The proposed transformer-based encoder is able to attend both intra-sentence and paper-level content.

Neural summarization of news documents. Several Deep Learning architectures have been proposed to extract summaries from news articles. For example, [8] propose a transformer-based binary classifier that predicts whether a news article sentence is worth including in the output summary. [9] and [10] propose similar approaches, where the extractive summarization problem is formulated as a sequence labeling task. More specifically, SummaRuNNer [10] exploits Recurrent Neural Networks to encode sentence content, whereas [9] propose a modular framework that enables the configuration of multiple encoder/decoder combinations. NeuSum [11] proposes a transformer-based model to jointly score and select sentences for news summarization. More recently, MatchSum [12] performs single-document summarization by first encoding both the source article and candidate summaries in the same latent space and then pick the summaries that are most similar to the reference article. In [12] the candidate summaries are generated using a transformer-based approach [8]. All the aforesaid summarization methods are not designed for extracting paper highlights. Furthermore, the summarization task is not formulated as a prediction of the Rouge-based sentence similarity score. The Refresh summarizer [13] relies on Reinforcement Learning. It exploits Rouge-based reward metrics [7] to select the most representative sentences. Despite the idea to reward the sentences that include the most similar content is the same, both the objective and the proposed Deep Learning architecture are substantially different.

Summarization of other document types. Extractive neural summarizers have been also exploited to summarize patent documents [14], biomedical documents [15], real-time events [16], and microblogging data [17]. The proposed approaches are focused on tailoring the summarization process to the peculiar characteristics of the input data. In this regard, the present work leverages

the attention mechanism to capture the inherent sentence-level relationships in the paper full-text.

3. Problem statement

Let $\mathcal{P}$ be a set of scientific papers (hereafter denoted as *training paper set*). Each paper $p \in \mathcal{P}$ is annotated with a set of K manually generated highlights $\mathcal{H}_p$ (typically, K is between 3 and 5). Given a paper p^* for which the corresponding annotations are unknown, highlights extraction aims at automatically generating the corresponding highlights $\mathcal{H}_{p^*}$.

This paper focuses on addressing highlights extraction by means of extractive summarization. Let $\mathcal{S}_{p^*}$ be an arbitrary set of sentence in p^*. The goal is to learn a scoring function $\mathcal{F} : \mathcal{S}_{p^*} \rightarrow \mathbb{R}^{[0,1]}$ from the training paper set that can be used to extract an output summary maximizing the similarity between the selected sentences and the expected highlight, i.e.,

$$\mathcal{H}_{p^*} = \underset{S \in \mathcal{P}(\mathcal{S}_p)}{\arg} \; max \, \mathcal{F}(S)$$

$$s.t.$$

$$|\mathcal{H}_{p^*}| \leq K$$

$$\mathcal{F} = sim(\mathcal{H}_{p^*}, S)$$

To this end, we model the relationship between the relevance score of an arbitrary sentence s^* in the test paper p^* and the content of the test sentence as an arbitrary regression function $\mathcal{F}_R$. Specifically, the target variable r_s is expressed as follows: $r_s = \mathcal{F}_R(s^*)$, where $f_R(\cdot)$ is the target prediction function, which is computed on the training paper set $\mathcal{P}$. The regression model is trained to minimize the mean square error between the predicted and expected similarity scores.

Similar to most recently proposed highlights extraction methods [1, 3], we quantify the function $\mathcal{F}_R$ as the syntactical Rouge similarity score between the contents of the candidate sentence and the paper highlights [7].

4. Rouge metrics

We exploit the Rouge toolkit [7] to compute the syntactical similarity between the (human-generated) paper highlights and the selected sentences. Rouge is the most commonly used metric to assess extractive summarization methods. It counts the unit overlaps between the extracted portion of text and the reference content (i.e., the overlap between the selected sentence and the expected highlights). Notice that the reference text does not necessarily match any of the paper sentences. Thus, the Rouge metrics typically returns partial overlap counts. Depending on the type of considered textual units, the following Rouge metrics are considered:

1. *Rouge-N*: it measures the overlap of *N-grams* between the generated highlights and the ground truth[2].
2. *Rouge-L*: it identifies the longest common sub-sequence of words.

The Rouge scores are quantified by the precision, recall, and F-measure values obtained by the summarization method. Specifically, recall@K is the ratio of correctly selected units in the top-K sentences to all the units in the reference highlights. It measures the ability to retrieve as much highlight units of text as possible. Precision@K indicates the percentage of the correctly selected units in the top-K sentences over all the units in the short-listed sentences. It quantifies the ability of the summarizer to accurately select relevant content. F-measure@K is the harmonic mean of precision and recall.

Since the goal of highlights extraction is to find the paper sentences whose content is most similar to that of the expected highlights, we adopt as target relevance score r_s the expected sentence similarity, which is computed as the maximum Rouge-2 F-measure score between the candidate sentence s^* and any of the reference highlights.

5. Transformer-based Highlights Extraction

The Transformer-based Highlights Extraction (THExt) architecture leverages contextualized embeddings and transformer models in the extraction of

[2]Throughout the paper we consider the unigrams (N=1) and bigrams (N=2), as they are the most commonly used in text summarization [7].

relevant paper highlights. THExt relies on the established BERT transformer model [6]. Thanks to its pretraining & fine-tuning paradigm, BERT has achieved impressive performance on several NLP tasks, including text summarization (see, for example [8]). It makes use of a bidirectional encoder representation, which relies on the attention mechanism to learn contextual relationships between words.

THExt leverages BERT capability to attend relevant information at the sentence level and to discover relevant patterns from the raw text (without the need for generating and exploring customized linguistic features). To overcome the limitations of BERT in handling longer pieces of text, THExt also integrates an alternative sentence encoder, namely LongFormer [19]. As discussed later on, the integration of LongFormer is instrumental for exploring the use of the paper full-text as additional context.

Figure 1 depicts the THExt architecture used for end-to-end training. It consists of the following steps: (1) Context definition, (2) Sentence encoding based on BERT, and (3) Regression based on a Fully Connected Network. The regression step computes the relevance scores of each candidate sentence and exploits them to drive highlights extraction.

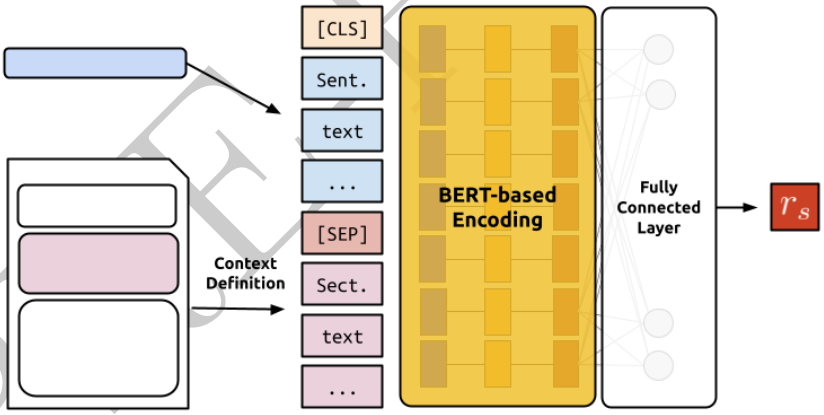


Figure 1: The presented architecture relying on transformer-based encoder.

Context definition. It selects a snippet of the paper p^*'s full-text, which is instrumental for contextualizing the sentence content and attending information relevant to highlights extraction. The paper context $\mathcal{C}$ contains what the model really needs to know in order to properly evaluate the candidate

sentences. Hence, the considered snippet of text is expected to include a discussion on the main findings of the research work presented in p^*.

To define the context $\mathcal{C}$ we leverage paper sectioning. Specifically, to parse the paper full-text at the section level, we apply regular expressions to the section title according to the IMRAD classification [18].

We explore the use of the following full-text snippets:

- *Abstract*: we consider it as a candidate context because it consists of a synthetic overview of the main research findings and meets the maximum context size constraint (i.e., 512 words minus the candidate sentence length).
- *Introduction*: it is deemed as an eligible context description because it commonly provides relevant insights into the main research findings. Unfortunately, in most cases, its content does not fit the maximum context size. Since the result-oriented discussions are usually placed at the end of the section we incrementally add the ending sentences of the introduction to the context until the maximum length is reached.
- *Results*: at the end of the results section, paper authors usually include comment on the empirical outcomes that are potentially worth considering for highlights extraction. Hence, we apply the same procedure previously described for the introductory section.
- *Conclusions*: they often include the takeaways of the research work, which can be suited to the highlights content.

Sentence encoding. We use the BERT model to encode the concatenation of the two following sequences (separated by a `[SEP]` token): (1) the content of the candidate sentence s^* and (2) the description of the paper context $\mathcal{C}$. The key idea is to capture not only the underlying local patterns in s^*, but also the global content dependencies summarized by $\mathcal{C}$. To this purpose, thanks to the attention mechanism [5], 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 model [5] computes the attention scores as follows:

$$Attention(Q, K, V) = softmax\Big(\frac{QK^T}{\sqrt{d_k}}\Big) \quad (1)$$

We compute the attention scores by considering both the local candidate sentence and the paper context. To overcome the BERT limitations related to the maximum number of input tokens (512) we use LongFormer [19] while considering, as reference context, the paper full-text. To this aim, we use a set of additional projections to add the global attention.

Regression. The encoded BERT sentences are provided as input to a fully connected neural network, which is used to train the regression model (see Section 3). In the training phase, the network minimizes the error between the predicted and expected relevance scores.

6. Experiments

6.1. Benchmark data

We run the experiments on the following benchmark datasets tailored to the highlights extraction problem and relative to different domains: (i) *CSPubSumm* [1], which collects scientific papers related to the Computer Science area, (ii) *BIOPubSumm*, which consists of papers from the biology and medicine domain, and (iii) *AIPubSumm* [3], which relates the Artificial Intelligence field. For each paper the dataset includes title, abstract, full-text, and 3 to 5 human-generated highlights.

The main dataset statistics are reported in Table 1. Notice that the per-dataset paper and sentence distributions are rather variable and representative of different scenarios. The average paper length in *AIPubSumm* is significantly higher than those in *CSPubSumm* and *BIOPubSumm*.

6.2. Baseline methods

We test the performance of four THExt variants relying on different contextual information, i.e.,

- *THExt-Abstract*: to encode the paper context it combines the target sentence with the abstract.
- *THExt-Intro*: it encodes both the target sentence and the introduction of the paper.

		#	Paper		Abstract		Introduction		Results		Conclusions	
			#W	#S	#W	#S	#W	#S	#W	#S	#W	#S
CS	train	10131	8236.63	262.26	316.81	10.61	4658.60	141.29	1818.11	65.72	413.59	14.01
	test	150	6010.78	196.08	297.91	10.22	1689.50	52.39	1486.82	47.53	376.60	11.69
BIO	train	8068	4894.24	160.75	371.04	13.23	1967.54	64.45	1559.76	49.43	272.52	10.77
	test	2690	4946.15	160.91	364.54	13.16	1957.52	63.55	1572.26	49.37	268.67	10.42
AI	train	198	10594.16	344.73	429.73	13.43	5372.88	167.12	2585.76	91.38	588.71	20.87
	test	66	11028.37	352.89	413.66	13.36	5506.45	162.18	1894.48	65.13	552.68	20.34

Table 1: Datasets' statistics. # represents the number of unique paper in the data collection. #W and #S represents the number of words and sentences respectively.

- *THExt-Results*: it contextualizes the sentence encoding using the experimental results.
- *THExt-Conclusions*: it contextualizes the sentence encoding using the conclusions of the paper.
- *THExt-FullText*: it encodes the full-text of the paper.
- *THExt-NoContext*: it encodes the target sentence solely, i.e., no additional context is provided.

For each THExt variants we test two different pre-trained BERT models: (1) the classical BERT model [6] trained on general-purpose documents (e.g., English-written Wikipedia) and (2) the SciBERT model [20], which is pre-trained on scientific documents.

As competitors we considered

1. The two state-of-the-art highlights extraction methods, namely [1] and [3], and
2. Two state-of-the-art supervised extractive summarization approaches not specifically designed to address highlight extraction, namely [9], [8].

The algorithms of category (1) comprise, to the best of our knowledge, the latest solutions to the highlights extraction process. The approaches of categories (2) are designed to solve a similar task, i.e., sentence-based summary extraction from news articles. Since this particular kind of summarizers are inherently portable to the highlights extraction domain we also explore their performance in such a new context.

Finally, since the problem of highlights extraction is supervised, hereafter we will not report the comparison with any unsupervised approach to text summarization because their scope is radically different. Based on the a set of preliminary experiments conducted on the same datasets, the performance of unsupervised methods has shown to be significantly worse.

6.3. Experiments' setup

Experiments were run on a machine equipped with AMD® Ryzen 9® 3950X CPU, Nvidia® RTX 3090 GPU, 128 GB of RAM running Ubuntu 20.04 LTS.

To perform end-to-end training of the domain-specific regression models we run one pre-training epoch on the union of all benchmark datasets (i.e., a large multiple-domain paper set) and then fine-tune the model for an additional epoch separately on each domain.

In our experiments we set the batch size to 32, the learning rate to 10^{-5} and we use the MSE loss [21] jointly with Adam optimizer [22]. In the evaluation phase we set the maximum lengths of the *candidate sentences* and of the *paper context* to 128 and 256, respectively. Using the aforesaid setting, the end-to-end training process took approximately 15 hours per domain.

To compare the highlights generated by different extractors we used (i) the Rouge metrics described in Section 4, which evaluate the highlights' similarity with the humanly generated highlights, and (ii) the Mean Reciprocal Rank (MRR), which evaluates the pertinence of the shortlisted sentences by decreasing Rouge score. It is computed as the harmonic mean of the ranks of the selected sentences in the ground truth. Unlike Rouge, it takes the order of appearance of the sentences in the summary into account [23].

For the existing new summarization algorithms [8, 9] and for [3] we fine-tuned the proposed model by exploiting the implementations provided by the respective authors, whereas for [1] we re-implemented the proposed method based on the description reported in the paper. The source code of the THExt architecture and pretrained models are freely available for research purposes[3].

[3] https://github.com/MorenoLaQuatra/THExt (latest access: May 2022)

6.4. Overall performance analysis

In Table 2 we compare the Rouge scores achieved by THExt with those of the competitors enumerated in Section 6.2. Here we focus on the performance of the best performing THExt variant (THExt-Abstract) using different pre-trained sentence encodings (i.e., BERT and SciBERT). The THExt performance is superior to that of all the tested competitor; e.g., *THExt-A*bstract with SciBERT +0.056 (+17%) against [3] on CSPubSumm, +0.028 (+10%) on BIOPubSumm, +0.035 (+12%) on AIPubSumm in terms of ROUGE-L F1-score. The improvements are all statistically significant according to the paired t-test [24] with 95% confidence level for every Rouge metrics and dataset.

Sentence ordering in THExt summaries meets the expected ranks with a significantly higher accuracy, e.g., MRR +0.094 (+105%) against [3] for AI domain. In Table 2 we also report similar results for the THExt version without paper contextualization (THExt-NoContext). By skipping the section-level content in the context encoding the highlights extraction was unable to attend the most discriminating patterns relevant to sentence ranking.

6.5. Further explorations

We separately analyze the effect of different characteristics of the proposed THExt architecture.

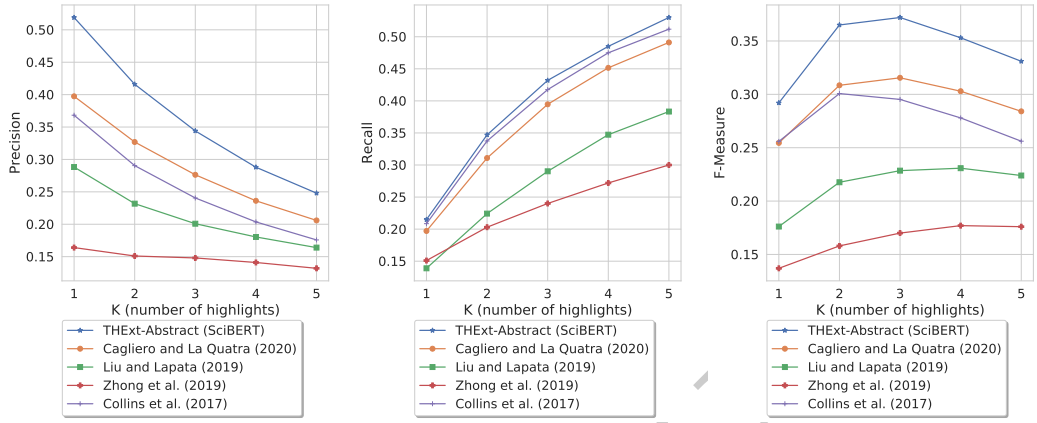
Effect of the number of selected highlights. Figures 2-4 show the effect of varying the number K of selected highlights on the extraction performance. As expected, recall values increase while increasing the number of selected highlights, whereas precision values show an opposite trend.

The majority of the extractors achieve the best F-measure score by setting K to 3. It turns out to be the best trade-off between result precision and coverage and is also coherent with the requested number of highlights in a real scenario (typically, between 3 and 5).

Effect of context selection. We analyze the effect of varying the definition of paper context on the THExt performance. Specifically, Table 3 compares the Rouge results achieved by THExt-NoContext (no paper context), THExt-Abstract, THExt-Intro, THExt-Results, and THExt-Conclusions. Concatenating the abstract content to the target sentence produces the best results, whereas ignoring the paper content yields the worst ones. The results are

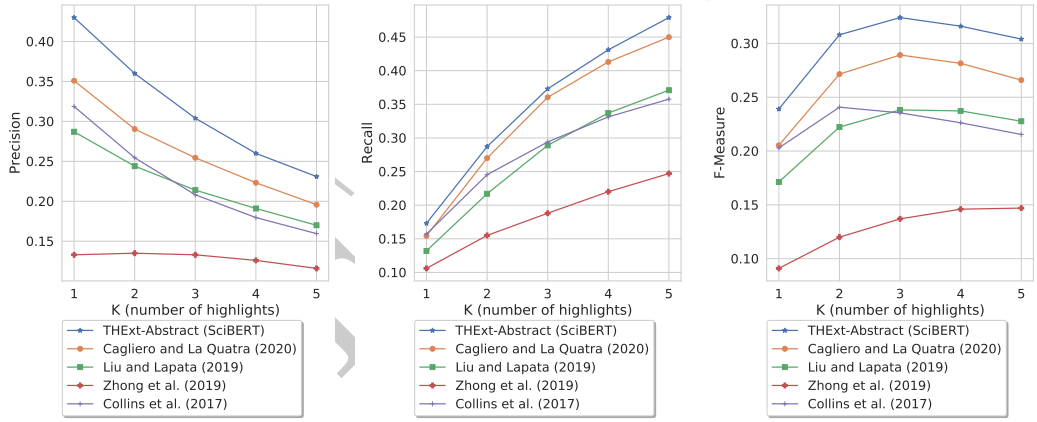
Metrics	R1	R2	RL	MMR
Computer Science				
Zhong et al., 2019 [9]	0.192*	0.031*	0.170*	-
Liu and Lapata, 2019 [8]	0.252*	0.058*	0.228*	-
Collins et al., 2017 [1]	0.339*	0.127*	0.295*	0.136*
Cagliero and La Quatra, 2020 [3]	0.364*	0.139*	0.316*	0.151*
THExt-NoContext (BERT)	0.354*	0.141*	0.326*	0.175*
THExt-NoContext (SciBERT)	0.352*	0.142*	0.325*	0.170*
THExt-Abstract (BERT)	0.392	0.184	0.362	**0.269**
THExt-Abstract (SciBERT)	**0.399**	**0.192**	**0.372**	0.267
Biology and Medicine				
Zhong et al., 2019 [9]	0.192*	0.033*	0.172*	-
Liu and Lapata, 2019 [8]	0.249*	0.059*	0.224*	-
Collins et al., 2017 [1]	0.287*	0.087*	0.243*	0.079*
Cagliero and La Quatra, 2020 [3]	0.316*	0.112*	0.280*	0.133*
THExt-NoContext (BERT)	0.315*	0.105*	0.286*	0.153*
THExt-NoContext (SciBERT)	0.316*	0.107*	0.288*	0.157*
THExt-Abstract (BERT)	0.337	0.126	0.307	0.198
THExt-Abstract (SciBERT)	**0.338**	**0.127**	**0.308**	**0.203**
Artificial Intelligence				
Zhong et al., 2019 [9]	0.152*	0.022*	0.137*	-
Liu and Lapata, 2019 [8]	0.266*	0.059*	0.238*	-
Collins et al., 2017 [1]	0.279*	0.069*	0.235*	0.069*
Cagliero and La Quatra, 2020 [3]	0.334*	0.111*	0.289*	0.089*
THExt-NoContext (BERT)	0.314*	0.103*	0.283*	0.122*
THExt-NoContext (SciBERT)	0.316*	0.104*	0.287*	0.135*
THExt-Abstract (BERT)	0.340	0.127	0.313	0.167
THExt-Abstract (SciBERT)	**0.355**	**0.137**	**0.324**	**0.183**

Table 2: Performance comparison, in terms of Rouge-1 (R1), Rouge-2 (R2), Rouge-L (RL) F-measure scores and MRR score on the benchmark datasets. K=3. Statistical relevant improvements between the best performing THExt extractor and the other methods are starred (*). The MRR scores of [8] and [9] are not specified since they do not provide any explicit sentence rank.



(a) Precision@K (b) Recall@K (c) F-Measure@K

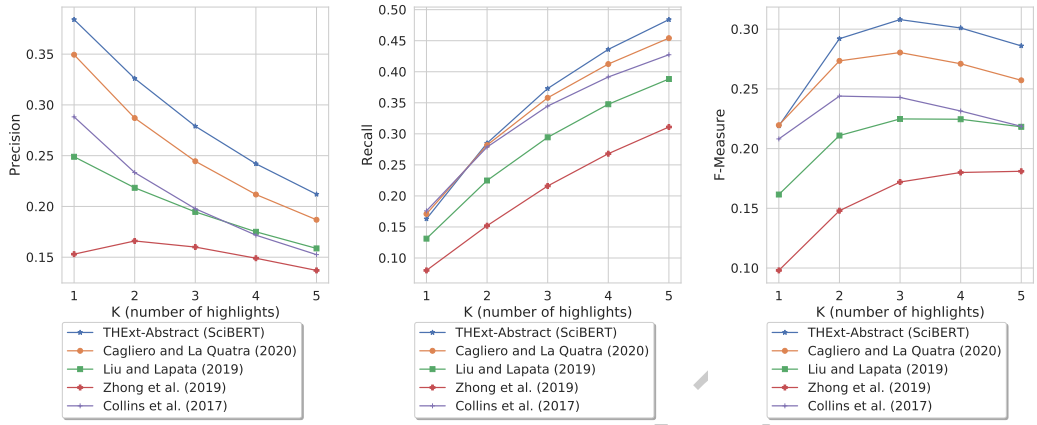
Figure 2: CSPubSumm dataset: effect of the number of extracted highlights. Rouge-L F-measure.



(a) Precision@K (b) Recall@K (c) F-Measure@K

Figure 3: AIPubSumm dataset: effect of the number of extracted highlights. Rouge-L F-measure.

coherent with the preliminary outcomes achieved by [1], which foster the exploration of the abstract content to accurately extract the paper highlights. Notably, the contextualization based on the results section achieves poor performance on CSPubSumm and BIOPubSumm and fair ones on AIPubSumm even if the highlights are expected to be result-oriented. This is probably



(a) Precision@K (b) Recall@K (c) F-Measure@K

Figure 4: BIOPubSumm dataset: effect of the number of extracted highlights. Rouge-L metrics.

due to the presence of special characters (e.g., numbers, symbols, acronyms) in the results' description, which are less likely to appear in the abstract and in the introduction.

In Table 3 we also explore different context settings for the proposed method. Specifically, to overcome the limitations of traditional transformer-based encoders like BERT [6] in the maximum number of input tokens, we also test a variant of the model, namely THExt-FullText, that is based on LongFormer [19]. The idea is to expand the context window beyond the standard maximum token number (512). The achieved results show that the performance of THExt-FullText is lower than expected. The reason is that LongFormer suffers from the windowed-attention mechanism. While providing the encoder with the paper full-text, LongFormer was unable to effectively estimate long-span attention scores.

Effect of the expected similarity score. We also evaluate the best performing model (i.e., THExt-Abstract) using a different regression objective. The goal here it to investigate the use of an alternative target function. Instead of using the Rouge-based overlap, previously considered in [3, 1], we adopt as target score the semantic similarity of each sentence with the manually annotated highlights. Specifically, we adopt the best performing Sentence-BERT

model [25][4] trained on the Semantic Textual Similarity task to compute the scores. The semantics-based model performs significantly worse than all the Rouge-based ones. This confirms the effectiveness of syntactic similarity scores in extractive summarization.

Effect of model pretraining. The quality of the unsupervised pretraining phase relevantly influences the accuracy of the regression models. Tailoring the pretrained models to documents belonging to the domain under consideration is known to be beneficial in many NLP tasks [4]. As expected, the results achieved using the SciBERT pretrained model [20] are, on average, superior to those obtained by the general-purpose pretrained vectors [6] (see Table 2).

6.6. Qualitative evaluation

In Table 4 we compare the highlights extracted by the THExt architecture with the reference highlights as well as with those of the state-of-the-art competitors [1, 3]. Unlike the highlights generated by [1], those produced by THExt appear more focused (e.g., they mention handgrip and dexterity). Compared to the output produced by [3], sentences appear to be more similar to the expected highlights.

THExt can be applied to scientific papers of different formats and relative to various domains. To allow readers to examine a broader set of results we generated and released[5] the highlights extracted from

- All the 2021 articles published on the open-access Journal of Machine Learning Research (JMLR Volume 22, 2021)[6]
- All the 2021 conference papers accepted for publication at Association for Computational Linguistics (ACL 2021)[7]

To extract the source text we exploited GROBID [26]. Notice that the results were generated without applying further post-processing step. For conference papers we did apply any fine-tuning step. The achieved results

[4] `https://www.sbert.net/docs/usage/semantic_textual_similarity.html` (latest access: May 2022) - `all-MiniLM-L6-v2` model tag

[5] https://github.com/MorenoLaQuatra/THExt (latest access: December 2021)

[6] https://www.jmlr.org/papers/v22/ (latest access: December 2021)

[7] https://aclanthology.org/volumes/2021.acl-long/ (latest access: December 2021)

Metrics	R1	R2	RL	MMR
Computer Science				
THExt-NoContext	0.352*	0.142*	0.325*	0.170*
THExt-Intro	0.359*	0.147*	0.33*	0.186*
THExt-Results	0.355*	0.146*	0.329*	0.182*
THExt-Conclusions	0.337*	0.135*	0.312*	0.131*
THExt-FullText†	0.225*	0.035*	0.202*	0.036*
THExt-Abstract-Semantic	0.307*	0.106*	0.28*	0.105*
THExt-Abstract	**0.399**	**0.192**	**0.372**	**0.267**
Biology and Medicine				
THExt-NoContext	0.316*	0.107*	0.288*	0.156*
THExt-Intro	0.334*	0.119*	0.303*	0.176*
THExt-Results	0.325*	0.114*	0.295*	0.182*
THExt-Conclusions	0.299*	0.096*	0.272*	0.132*
THExt-FullText†	0.167*	0.032*	0.148*	0.045*
THExt-Abstract-Semantic	0.274*	0.081*	0.248*	0.104*
THExt-Abstract	**0.338**	**0.127**	**0.308**	**0.203**
Artificial Intelligence				
THExt-NoContext	0.316*	0.104*	0.287*	0.136*
THExt-Intro	0.326*	0.109*	0.294*	0.150*
THExt-Results	0.333*	0.112*	0.303*	0.155
THExt-Conclusions	0.289*	0.084*	0.261*	0.091*
THExt-FullText†	0.18*	0.027*	0.161*	0.043*
THExt-Abstract-Semantic	0.295*	0.09*	0.266*	0.107
THExt-Abstract	**0.355**	**0.137**	**0.324**	**0.183**

Table 3: Performance comparison, in terms of Rouge-1 (R1), Rouge-2 (R2), Rouge-L (RL) F-measure scores and MRR score between different THExt variants. K=3. SciBERT pretrained model. Statistical relevant improvements between the best performing method (THExt-Abstract) and the other variants are starred (*). The model marked with † is based on [19].

show the generality of the proposed approach, as it can be successfully applied to papers ranging over different topics, published on various venues, and characterized by variable structures.

7. Conclusions and future work

In this paper we propose a new approach, based on transformer encoding, to extract highlights from scientific papers. The architecture conveniently combines the content of the target sentence with a paper context consisting of the content of the paper sections. The use of the attention mechanism has proved to be beneficial for training supervised regressors in different domains. Attending the abstract content has shown to be particularly effective in capturing the underlying patterns in the analyzed text, whereas using sections other than the abstract as reference context (e.g., introduction, conclusions) do not provide relevant performance improvements. The exploration of more complex Transformer-based encoder, such as [19], allowed us to investigate the use of the paper full-text as contextual information. However, the inherent limitations of the model in estimating long-span attention scores hinders significant performance improvements.

As future work, we plan to drill down through the sentence content and operate at the character level by exploiting the token-free transformer models [28]. The goal is to effectively handle noisy characters, which likely influence the performance of the models integrating results content, especially in the biological domain. Furthermore, we will investigate the use of attention-focused methods for abstractive summarization, such as the one recently proposed by [29].

References

[1] E. Collins, I. Augenstein, S. Riedel, A supervised approach to extractive summarisation of scientific papers, in: Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017), 2017, pp. 195–205.

[2] Elsevier, Highlights: author information (2021).
URL `https://www.elsevier.com/authors/tools-and-resources/highlights`

Table 4: Qualitative comparison between the automatically generated and the manually annotated highlights of the paper entitled *Design options for improving protective gloves for industrial assembly work* [27].

Humanly generated highlights
1) Glove design 1 and barehanded produced higher handgrip strength than double gloves.
2) Glove design 1 and the bare hand condition had better dexterity than double gloves.
3) Selective thickness over some hand areas can be applied to improve the glove design.
THExt
1) Also, glove design 1 and the bare hand condition produced higher handgrip strength values compared to double gloves.
2) The findings support the hypothesis that selective variation in thickness over some areas of the hand can be applied to improve industrial glove design.
3) The results of the study indicated that the bare hand condition and glove design 1 produced shorter pegboard completion times (which means better dexterity) than double gloves.
Cagliero and La Quatra (2020)
1) The findings support the hypothesis that selective variation in thickness over some areas of the hand can be applied to improve industrial glove design.
2) The present study investigated the effect of the two modified designs of glove on a number of hand performance capabilities and subjective assessments and also compared their performance against those with bare hand, single-layered cotton glove and double cotton glove while working with pliers and a screwdriver.
3) It appears that improved dexterity and tactile feedback in some areas of the hand with the first prototype glove tended to give more improvements in performance and strength capabilities of the hand than with the second prototype glove.
Collins et al. (2017)
1) Their effects on a number of hand performance capabilities were measured and the wearers' subjective assessments were collected.
2) One was based on published evidence of the differences in the areas of discomfort between the bare hand and the gloved hand conditions, and the other was based on the need for finger and hand flexibility while working with tools such as screwdriver and pliers.
3) Handgrip strength, lateral pinch strength and forearm torque strength were three different maximal isometric strength measurements taken during the experiment.

[3] L. Cagliero, M. La Quatra, Extracting highlights of scientific articles: A supervised summarization approach, Expert Systems with Applications

160 (2020) 113659.

[4] J. Camacho-Collados, M. T. Pilehvar, Embeddings in natural language processing, in: Proceedings of the 28th International Conference on Computational Linguistics: Tutorial Abstracts, International Committee for Computational Linguistics, Barcelona, Spain (Online), 2020, pp. 10–15. doi:10.18653/v1/2020.coling-tutorials.2.
URL https://www.aclweb.org/anthology/2020.coling-tutorials.2

[5] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, u. Kaiser, I. Polosukhin, Attention is all you need, in: Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS'17, Curran Associates Inc., Red Hook, NY, USA, 2017, p. 6000–6010.

[6] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 4171–4186. doi:10.18653/v1/N19-1423.
URL https://www.aclweb.org/anthology/N19-1423

[7] C.-Y. Lin, Rouge: A package for automatic evaluation of summaries, in: Text summarization branches out, 2004, pp. 74–81.

[8] Y. Liu, M. Lapata, Text summarization with pretrained encoders, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), 2019, pp. 3721–3731.

[9] M. Zhong, P. Liu, D. Wang, X. Qiu, X. Huang, Searching for effective neural extractive summarization: What works and what's next, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Florence, Italy, 2019, pp. 1049–1058. doi:10.18653/v1/P19-1100.
URL https://www.aclweb.org/anthology/P19-1100

[10] R. Nallapati, F. Zhai, B. Zhou, Summarunner: A recurrent neural network based sequence model for extractive summarization of documents, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 31, 2017.

[11] Q. Zhou, N. Yang, F. Wei, S. Huang, M. Zhou, T. Zhao, A joint sentence scoring and selection framework for neural extractive document summarization, IEEE/ACM Trans. Audio, Speech and Lang. Proc. 28 (2020) 671–681. doi:10.1109/TASLP.2020.2964427.
URL https://doi.org/10.1109/TASLP.2020.2964427

[12] M. Zhong, P. Liu, Y. Chen, D. Wang, X. Qiu, X. Huang, Extractive summarization as text matching, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Online, 2020, pp. 6197–6208. doi:10.18653/v1/2020.acl-main.552.
URL https://www.aclweb.org/anthology/2020.acl-main.552

[13] S. Narayan, S. B. Cohen, M. Lapata, Ranking sentences for extractive summarization with reinforcement learning, in: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), Association for Computational Linguistics, New Orleans, Louisiana, 2018, pp. 1747–1759. doi:10.18653/v1/N18-1158.
URL https://www.aclweb.org/anthology/N18-1158

[14] M. Zaheer, G. Guruganesh, A. Dubey, J. Ainslie, C. Alberti, S. Ontanon, P. Pham, A. Ravula, Q. Wang, L. Yang, A. Ahmed, Big bird: Transformers for longer sequences (2021). arXiv:2007.14062.

[15] Y. Du, Q. Li, L. Wang, Y. He, Biomedical-domain pre-trained language model for extractive summarization, Knowledge-Based Systems 199 (2020) 105964.

[16] M. Yang, C. Li, F. Sun, Z. Zhao, Y. Shen, C. Wu, Be relevant, non-redundant, and timely: Deep reinforcement learning for real-time event summarization, Proceedings of the AAAI Conference on Artificial Intelligence 34 (05) (2020) 9410–9417. doi:10.1609/aaai.v34i05.6483.
URL https://ojs.aaai.org/index.php/AAAI/article/view/6483

[17] A. Dusart, K. Pinel-Sauvagnat, G. Hubert, Issumset: A tweet summarization dataset hidden in a trec track, in: Proceedings of the 36th Annual ACM Symposium on Applied Computing, SAC '21, Association for Computing Machinery, New York, NY, USA, 2021, p. 665–671. doi:10.1145/3412841.3441946.
URL https://doi.org/10.1145/3412841.3441946

[18] L. Sollaci, M. Pereira, The introduction, methods, results, and discussion (imrad) structure: A fifty-year survey, Journal of the Medical Library Association 92 (3) (2004) 364–367.

[19] I. Beltagy, M. E. Peters, A. Cohan, Longformer: The long-document transformer, arXiv:2004.05150 (2020).

[20] I. Beltagy, K. Lo, A. Cohan, SciBERT: A pretrained language model for scientific text, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, Hong Kong, China, 2019, pp. 3615–3620. doi:10.18653/v1/D19-1371.
URL https://www.aclweb.org/anthology/D19-1371

[21] C. Sammut, G. I. Webb (Eds.), Mean Squared Error, Springer US, Boston, MA, 2010, pp. 653–653. doi:10.1007/978-0-387-30164-8_528.
URL https://doi.org/10.1007/978-0-387-30164-8_528

[22] I. Loshchilov, F. Hutter, Decoupled weight decay regularization, in: 7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019, OpenReview.net, 2019.
URL https://openreview.net/forum?id=Bkg6RiCqY7

[23] C. D. Manning, P. Raghavan, H. Schütze, Introduction to Information Retrieval, Cambridge University Press, USA, 2008.

[24] J. Pfanzagl, O. Sheynin, Studies in the history of probability and statistics xliv a forerunner of the t-distribution, Biometrika (1996) 891–898.

[25] N. Reimers, I. Gurevych, Sentence-bert: Sentence embeddings using siamese bert-networks, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing, Association for Com-

putational Linguistics, 2019.
URL http://arxiv.org/abs/1908.10084

[26] P. Lopez, Grobid: Combining automatic bibliographic data recognition and term extraction for scholarship publications, in: International conference on theory and practice of digital libraries, Springer, 2009, pp. 473–474.

[27] I. Dianat, C. M. Haslegrave, A. W. Stedmon, Design options for improving protective gloves for industrial assembly work, Applied Ergonomics 45 (4) (2014) 1208–1217. doi:https://doi.org/10.1016/j.apergo.2014.02.009.
URL https://www.sciencedirect.com/science/article/pii/S0003687014000301

[28] L. Xue, A. Barua, N. Constant, R. Al-Rfou, S. Narang, M. Kale, A. Roberts, C. Raffel, Byt5: Towards a token-free future with pretrained byte-to-byte models, arXiv preprint arXiv:2105.13626 (2021).

[29] X. Duan, H. Yu, M. Yin, M. Zhang, W. Luo, Y. Zhang, Contrastive attention mechanism for abstractive sentence summarization, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), Association for Computational Linguistics, Hong Kong, China, 2019, pp. 3044–3053. doi:10.18653/v1/D19-1301.
URL https://www.aclweb.org/anthology/D19-1301