USING TRANSFORMERS TO MULTI-LABEL LONG LEGAL DOCUMENTS

Hanane el Aajati / Roderick Lucas / Radboud Winkels

Hanane el Aajati, FNWI, University of Amsterdam, Netherlands, h.aajati@uva.nl

Roderick Lucas, Deloitte, Amsterdam, Netherlands, RLucas@deloitte.nl

Radboud Winkels, PPLE College, University of Amsterdam, Roetersstraat 11, 1018 WB Amsterdam, Netherlands, winkels@uva.nl

Keywords: ***Transformers; multi-label classification; summarization; Longformer***

Abstract: *This paper describes experiments to find an efficient method to multi-label official legal documents using Transformers. Transformers are popular and powerful ML models because they are pre-trained on large amounts of data. A significant drawback is that most Transformers cannot process documents longer than 512 tokens. We try to tackle this issue by proposing a new method to multi-label long documents – summarizing long documents first before multi-labeling with the bert-base-cased Transformer. This summarization method is compared with two existing methods: truncating documents after the first 512 tokens and Longformer. The methods are evaluated on the F1 score, and the results show that the Longformer performs the best. The summarization method and truncating method seem to output almost equal F1 scores. Although the summarization method did not perform well on the dataset used in this research, it could be promising for datasets with more structured documents.*

1. Introduction

One of the first steps lawyers take when receiving a legal case to work on is finding documents describing similar situations. A method frequently used is searching for lawsuits with labels that define the content of the legal issue. Document collections are categorized according to their content using a set of labels generated from some type of thesaurus to increase structure and support easy access to relevant information. For instance, official documents published by the European Commission utilize the Eurovoc thesaurus[1]. The practice of assigning thesaurus labels to documents is typically done manually by a team of skilled documentalists. Automatic multi-labeling legal documents could make the process of assigning labels easier, more systematic, and less time consuming. This could increase these legal services' quality and efficiency and reduce the employee's workload. On the other hand, it may introduce more (systematic) errors than human experts would.

One approach to automate assigning labels to textual documents is using text classification techniques. Text classification includes methods such as binary classification, multi-class classification, and multi-label classification [1]. The latter uses a specialized machine learning algorithm so that multiple labels can be assigned to one document. For example, a legal document could be about both 'human rights' and 'tax law'.

Transformers are one of the most popular methods currently used for multi-labeling tasks [2]. The Transformer architecture allows for effective parallel training and scales with training data and model size. The purpose of a Transformer is to handle sequential input like textual data streams, and it uses an attention mechanism to detect relationships between sequential elements. One of the fundamental problems with Transformers is that the self-attention mechanism grows quadratically with sequence length. This results in the model not being able to process longer sequences. The current maximum amount most Transformer models can process is 512 tokens. This is a significant drawback because real-world data does not limit itself to this maximum.

[1] https://op.europa.eu/en/web/eu-vocabularies/dataset/-/resource

Researchers have proposed some methods to process longer documents. The most standard way is to truncate every token after the 512th; Meaning only the first 512 tokens are taken into account by the Transformer model [3]. Other approaches that have been researched are Transformers like Longformer [4] and Bird [5]. Although these types of Transformers provide a way to process longer documents, they have the disadvantage that they are expensive in their computing time. Another more recent approach is to minimize the document length by dividing the documents into chunks of 512 or less [6].

We propose a new method: Summarizing the document first to max 512 tokens before multi-labeling it. We will then compare these results with two existing methods; Longformer and truncating the document length after 512 tokens. The method of truncating the document length after 512 tokens will be referred to as 'truncating' and the method of summarizing before multi-labeling will be referred to as 'summarizing'. We expect the Longformer method to outperform the other methods and the summarizing method to outperform the truncating method.

2. Background

2.1 Bert

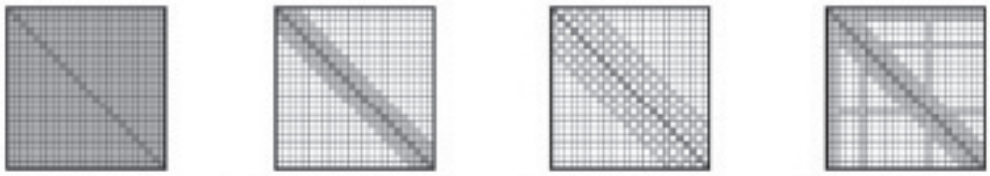
In 2018 Devlin et al. [7] introduced a Transformer model to represent language, Bidirectional Encoder Representations from Transformers (Bert). Bert uses a self-attention mechanism that grows quadratically with the sequence length. Figure 1a shows the self-attention pattern followed in Bert (and most other Transformers). Each token in this model attends the following-up token.

Bert is pre-trained on a large corpus containing English words using a self-supervised machine learning technique. This means that the model was trained without the help of humans manually labeling the data. This causes the model to be able to use tons of data that is publicly available. Bert has enjoyed unparalleled success in natural language processing thanks to two unique training approaches: masked-language modeling (MLM), and next sentence prediction (NSP). MLM ensures that the model randomly masks 15% of the input tokens. The masked input then goes through the model again, and the model now has the task of predicting the masked words. In NSP, Bert appends two masked sentences together as input during the pretraining phase. Sometimes, the appended sentences are also next to each other in the original text. Bert must determine whether the two sentences followed one another or not. Using these two features, Bert is able to acquire an internal representation of the English corpus. This can then be utilized to extract features for downstream tasks such as multi-label classification.

2.2 Longformer

As mentioned in the introduction, most Transformers use a self-attention mechanism that scales quadratically with the sequence length. Beltagy et al. [4] tackle this problem by proposing the Longformer, with an attention mechanism that scales linearly with sequence length. It employs three complementing patterns instead of the entire attention mechanism seen in most Transformer systems to support local as well as global attention:

1. Sliding window attention
2. Dilated sliding window attention
3. Global + sliding window attention



(a) Full n^2 attention (b) Sliding window attention (c) Dilated sliding window (d) Global + sliding window

Figure 1: Self-attention patterns [4]

Figure 1b shows the sliding window pattern which is used by the Longformer model. It relies on fixed-size windows to focus on each token. This helps a multi-layer transformer build a large receptive field in which the top layers can build representations that incorporate information from the entire input.

In addition, the Longformer architecture employs a pattern to dilate the sliding surface attention, as shown in Figure 1c. In this procedure, sliding windows are expanded with gaps of variable widths. This enables Longfrom to capture elements in the sequence far from each other in a single slide without adding to the computation.

Although sliding window patterns are great for forming local attention, they lack the versatility to establish task-specific representation; for this the larger context is needed. To overcome this difficulty, the Longformer architecture integrates a standard global attention mechanism for selected locations in the input (Figure 1d).

2.3 Summarization

For text summarization, one can choose two approaches: Extractive or abstractive [8]. Extractive summarization is the process of creating a summary based on the most important sentences of the original text document. It does this by selecting important words and sentences. The algorithm arranges them to create a summary. Extractive summarization can both be supervised and unsupervised. LexRank is an unsupervised technique in which the salience of the text is defined by the idea of eigenvector centrality [9].

The second approach for text summarization is abstractive. Abstractive text summarization is more similar to human summarizing. This involves advanced natural language processing and compression techniques. As a result, this is a computationally more intense task when compared to the extractive summarization technique, and we will not be using it.

2.4 The Data

For this research, a dataset provided by the moonlit.ai project team of Deloitte Belastingadviseurs BV is used. This dataset is scraped entirely from EUR-Lex (the official European Union law website). Every document has a unique identifier, the CELEX number, and may have one or more manually assigned labels from the Eurovoc multilingual thesaurus. The current edition of Eurovoc comprises almost 7,000 concepts pertaining to various EU and Member State activities.

Each document is divided into three key sections: the header, which comprises the title and name of the enforcing legal entity; the citations, which are legal background references; and the main content, which is normally organized in paragraphs.

The dataset contains 257,816 entries with unique 7,033 labels. We decided to reduce the number of labels to the most frequently used ones, decreasing data sparsity and achieving higher accuracy. Included labels appeared between 5,000–15,000 times in the dataset. This accounts for a total amount of 91 labels for a set

of 191.253 documents. The average document length is 2,567 tokens. This is considered a long document for most Transformers like Bert.

In the experiments described below we will see to what extent the various multi-labeling methods are able to find the same labels that the human experts attached to the documents.

3. The Experiment

The first method to multi-label the documents was only to use the first 512 tokens of each document ('truncated'). The dataset was divided into three independent sets:

1. A training set of 122,401 documents is used to train and discover any hidden patterns and features in the data. The same training data is supplied to the model repeatedly in each epoch, and the model continues to learn the data's features.
2. A validation set of 30,601 entries was used to validate the performance of the Transformer during training.
3. A test set of 38,251 entries to give the final performance metrics.

The Bert Transformer was used to multi-label the data. The parameters chosen for training are 12 for epochs and 8 for batch size. The general rule of thumb is to start with an epoch value of three times the number of columns in your dataset, which in this case is 3*3=9. After trying a few different values, it was determined that 12 resulted in the highest accuracy, and the model was then trained using these parameters.

The next method chosen is to summarize the long documents to 512 tokens for the Bert Transformer to process it. Since the dataset does not contain any labeled summaries, we had to use an unsupervised method. We tried out several and ended up using Gensim[2], which gave similar results as the others but with the best time performance. Next, we repeated the experiment with Bert as described above.

The final method chosen to compare the other multi-label methods is Longformer. The first step was to prepare the data labels for the Longformer. This is done by using one hot encoding (ohe). This is the process of turning categorical labels into binary integers where the value of 1 of a feature corresponds to the original label. The dataset was then split into a train, validation and test set as described above. The Longformer was also trained using 12 epochs and a batch size of 8.

4. Results

The precision, recall, and F1 score were used to evaluate the performance of the methods. Table 1 presents performance metrics for the three multi-labeling methods. We will use the F1 scores for our discussion, since the dataset is highly imbalanced. For any document in the dataset 91 possible labels could be assigned. In total, 82,209 times a label was assigned to a document, and 3,398,632 times a label was not assigned.

Table 1: Metric summary of the three multi-labeling methods

Method	Precision	Recall	F1 Score
Truncating	26%	6%	0,09
Summarizing	5%	15%	0,08
Longformer	71%	51%	0,59

The truncating method and the summarizing methods have almost equal F1 scores. Initially, we expected the summarization method to outperform the truncating method. It did not. Perhaps the Gensim summarizer did not perform as well since most of the documents are very unstructured; some don't have any interpunction, which

[2] https://radimrehurek.com/gensim/index.html

causes Gensim to see it as one big sentence. Also, the truncating method performs better than expected, which may be because quite a number of documents turn out to start with a short summary of the entire document.

The Longformer has an F1 score of almost 7 times as high as the other two methods. With a precision of 0.71, it was able to predict a label correctly 71 out of 100 times, but it only found 51 out of 100 labels that humans assigned. Of course, it may be that some of the labels the algorithms 'falsely' assigned, are in fact correct. We did not perform a manual analysis of the false positives or false negatives.

5. Conclusions and Future Work

We aimed to find a method to multi-label long documents within the legal domain with Transformers, tackling the limitation of 512 tokens these models can process. A new method was introduced to address this issue: summarizing the documents to 512 tokens before multi-labeling them with the Bert Transformer. This method was then compared with two existing methods to multi-label longer documents with Transformers: Truncating the data after 512 tokens and Longformer. Results show that Longformer outperforms the other methods.

Although the summarization method did not perform well on the EUR-lex dataset, it could be promising for datasets with more standardized documents.

Bert Transformer results could probably be improved if it were fine-tuned on legal data. Currently there is no such Longformer model, but there is LEGAL-BERT, a Transformer (limited to 512 tokens) that is. Legal-Bert only works for English texts unfortunately [10]. That is why Deloitte is using an in-house finetuned XLM-RoBertA model [11] for multilingual legal documents.

Another interesting approach to increase performance on the summarizing method is to use Longformer to summarize the documents. This is because the Longformer is pre-trained which could increase the accuracy of the summaries. Additionally, chunk-wise summarization could be interesting to research, where a document of N tokens is divided in N/512 chunks of 512 tokens (or some other small number), which are all summarized to 512^2/N tokens and finally all these summaries are appended into the final summary of 512 tokens.

6. References

[1] Grigorios Tsoumakas and Ioannis Katakis. "Multi-label classification: An overview". In: *International Journal of Data Warehousing and Mining* (IJDWM) 3.3 (2007), pp. 1–13.

[2] Ashish Vaswani et al. "Attention is all you need". In: *Advances in neural information processing systems* 30 (2017).

[3] Qizhe Xie et al. "Unsupervised data augmentation for consistency training". In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 6256–6268.

[4] Iz Beltagy, Matthew E Peters, and Arman Cohan. "Longformer: The long-document Transformer". In: *arXiv preprint arXiv:2004.05150* (2020).

[5] Manzil Zaheer et al. "Big bird: Transformers for longer sequences". In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 17283–17297.

[6] Mandar Joshi et al. "Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension". In: *arXiv preprint arXiv:1705.03551* (2017).

[7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova. "Bert: Pre-training of deep bidirectional Transformers for language understanding". In: *arXiv preprint arXiv:1810.04805* (2018).

[8] Udo Hahn and Inderjeet Mani. "The challenges of automatic summarization". In: Computer 33.11 (2000), pp. 29–36.

[9] Günes Erkan and Dragomir R Radev. "Lexrank: Graph-based lexical centrality as salience in text summarization". In: *Journal of artificial intelligence research 22* (2004), pp. 457–479.

[10] Ilias Chalkidis et al. "LEGAL-BERT: The muppets straight out of law school". In: *arXiv preprint arXiv:2010.02559* (2020).

[11] Yinhan Liu et al. "RoBERTa: A Robustly Optimized BERT Pretraining Approach". In *arXiv: arXiv:1907.11692* (2019).