LaTeX Author Guidelines for CVPR Proceedings

First Author
Institution1
Institution1 address
firstauthor@i1.org

Second Author
Institution2
First line of institution2 address
secondauthor@i2.org

Abstract

The ABSTRACT is to be in fully-justified italicized text, at the top of the left-hand column, below the author and affiliation information. Use the word "Abstract" as the title, in 12-point Times, boldface type, centered relative to the column, initially capitalized. The abstract is to be in 10-point, single-spaced type. Leave two blank lines after the Abstract, then begin the main text. Look at previous CVPR abstracts to get a feel for style and length.

1. Introduction

Please follow the steps outlined below when submitting your manuscript to the IEEE Computer Society Press. This style guide now has several important modifications (for example, you are no longer warned against the use of sticky tape to attach your artwork to the paper), so all authors should read this new version.

2. Current methods

2.1. RNN

2.2. Transformers

3. BertTextGenerator

3.1. Attention method

3.2. Finetuning

BERT is a powerful model pre-trained on a broad corpus composed by the whole Wikipedia and Brown corpora. Typical fine-tuning techniques depends on the final task that the user wants BERT to perform. For example, a classical application of the model is to predict the sentiment of a sentence. In this case the focus of the fine-tuning would be on the [CLS] token, a special token specifically used for classification tasks.

In this case however the task is different and unusual for BERT. Since we need to use BERT to generate text it is important that during the fine-tuning the model understands the structure of the text.

We have implemented a fine-tuning method that gave completely freedom to the user to decide the language, the structure and even the sentiment of the text to generate. The fine-tuning is performed considering only one task of the two original used to pre-train BERT; that is the mask-prediction. 15% of the tokens of each sentence are replaced with a masked token and BERT have to predict the original tokens. The loss is computed as the cross-entropy between the logits for the masked tokens outputted by BERT and the original tokens.

This method allows the final user to start from a pre-trained model or, if it is not available for the language chose by the user, from a cross-language model and fine-tuning on a specific text corpus.

The fine-tuning allows also to exploit the enormous potentialities of BERT tokenizer. The default vocabulary of the tokenizer contains 1000 unused tokens, whose weights are randomly initialized. Typically, these tokens are replaced with domain specific words, so that during the fine-tuning the model will be able to learn them. We have extended this idea in order to comprehend also tokens that defines the structure of the text like '
n' '
t'. Note that replacing in the vocabulary for example the token '[unused1]' with '
n' would have no effect since the tokenizer would remove these tokens from the text even before the tokenization, during the normalization step. To solve this problem we have implemented a Formatter class that helps the user mantaining a map between some user specified tokens that needs to be preserved and some unused tokens choosed to replace them. The user-specified tokens are replaced with the unused tokens before the tokenization and the finetuning and then are replaced back after the text generation. Using this method we were able to make a model learn the tercet structure of Dante's Divine Comedy.

3.2.1 Sentiment generation

BERT tokenizer gave also the possibility to define some special tokens. We have took advantage of this to define a new method to generate text with a specific sentiment. Considering a set of sentences each one with a possible sentiment labels in the set ['pos', 'neg'], we define n (3 by default) new special tokens for each possible sentiment. The special tokens are of the type [pos-i] $i = 1, ldots, n$. Before the finetuning the special tokens corresponding to the sentence sentiment label are appended at the beginning of the sentence. In this manner the model, during the finetuning is able to build a relations among the special tokens of a sentiment and the specific words related to that sentiment. At inference time, to generate some text with a specific sentiment, we simply pass the special tokens of that sentiment as seed_text of the generae method.

This method, even in it's simplicity, showed it's efficacy in generating positive and negative tweets in italian about football

4. Experiments and evaluation

5. Conclusions

References