EXPERIMENTAL RESULTS

Evaluate several configurations of the proposed model, namely three different depths and three different pooling types.

The main contribution is a thorough evaluation of networks of increasing depth using an architecture with small temporal convolution filters with different types of pooling, which shows that a significant improvement on the state-of-the-art configurations can be achieved on text classification tasks by pushing the depth to 29 convolutional layers.

Deep architecture works well on big data sets in particular, even for small depths. For the smallest depth we use (9 convolutional layers), the model already performs better than Zhang’s convolutional baselines (which includes 6 convolutional layers and has a different architecture) on the biggest data sets. The most important decrease in classification error can be observed on the largest data set Amazon Full which has more than 3 Million training samples.

No data preprocessing or augmentation is used.

For a small depth, temporal max-pooling works best on all data sets. Depth improves performance. As the network depth is increased to 17 and 29, the test errors decrease on all data sets, for all types of pooling (with 2 exceptions for 48 comparisons).

Going from depth 9 to 17 and 29 for Amazon Full reduces the error rate by 1% absolute. Since the test is composed of 650K samples, 6.5K more test samples have been classified correctly.

These improvements, especially on large data sets, are significant and show that increasing the depth is useful for text processing. Overall, compared to previous state-of-the-art, the best architecture with depth 29 and max-pooling has a test error of 37.0 compared to 40.43%.

This represents a gain of 3.43% absolute accuracy. The significant improvements which we obtain on all data sets compared to Zhang’s convolutional models do not include any data augmentation technique.

Max-pooling performs better than other pooling types. Both pooling mechanisms perform a max operation which is local and limited to three consecutive tokens, while k-max polling considers the whole sentence at once.

According to the experiments, it seems to hurt performance to perform this type of max operation at intermediate layers (with the exception of the smallest data sets). The models outperform state-of-the-art ConvNets.

However, with the proposed deep architecture, we get closer to the state-of-the-art which are ngrams TF-IDF for these data sets. As observed in previous work, differences in accuracy between shallow (TF-IDF) and deep (convolutional) models are more significant on large data sets, but we still perform well on small data sets while getting closer to the non convolutional state-of-the-art results on small data sets.

The very deep models even perform as well as ngrams and ngrams-TFIDF respectively on the sentiment analysis task of Yelp Review Polarity and the ontology classification task of the DBPedia data set. Results of Yang et al. (only on Yahoo Answers and Amazon Full) outperform our model on the Yahoo Answers dataset, which is probably linked to the fact that their model is task-specific to datasets whose samples that contain multiple sentences like (question, answer). They use a hierarchical attention mechanism that apply very well to documents (with multiple sentences).

Going even deeper degrades accuracy. Shortcut connections help reduce the degradation. The gain in accuracy due to the the increase of the depth is limited when using standard ConvNets. When the depth increases too much, the accuracy of the model gets saturated and starts degrading rapidly. This degradation problem was attributed to the fact that very deep models are harder to optimize. The gradients which are backpropagated through the very deep networks vanish and SGD with momentum is not able to converge to a correct minimum of the loss function. To overcome this degradation of the model, the ResNet model introduced shortcut connections between convolutional blocks that allow the gradients to flow more easily in the network (He et al., 2016a). We evaluate the impact of shortcut connections by increasing the number of convolutions to 49 layers. We present an adaptation of the ResNet model to the case of temporal convolutions for text (see Figure 1). Table 6 shows the evolution of the test errors on the Yelp Review Full data set with or without shortcut connections. When looking at the column “without shortcut”, we observe the same degradation problem as in the original ResNet article: when going from 29 to 49 layers, the test error rate increases from 35.28 to 37.41 (while the training error goes up from 29.57 to 35.54). When using shortcut connections, we observe improved results when the network has 49 layers: both the training and test errors go down and the network is less prone to underfitting than it was without shortcut connections. While shortcut connections give better results when the network is very deep (49 layers), we were not able to reach state-of-the-art results with them. We plan to further explore adaptations of residual networks to temporal convolutions as we think this a milestone for going deeper in NLP. Residual units (He et al., 2016a) better adapted to the text processing task may help for training even deeper models for text processing, and is left for future research. Exploring these models on text classification tasks with more classes sounds promising. Note that one of the most important difference between the classification tasks discussed in this work and ImageNet is that the latter deals with 1000 classes and thus much more information is back-propagated to the network through the gradients. Exploring the impact of the depth of temporal convolutional models on categorization tasks with hundreds or thousands of classes would be an interesting challenge and is left for future research.

CONCLUSION

New architecture for NLP which follows two design principles:

1) operate at the lowest atomic representation of text, i.e. characters

2) use a deep stack of local operations, i.e. convolutions and max-pooling of size 3, to learn a high-level hierarchical representation of a sentence.

The architecture was evaluated on eight freely available large-scale data sets which showed that increasing the depth up to 29 convolutional layers steadily improves performance. The new models are much deeper than previously published convolutional neural networks and they outperform those approaches on all data sets. Even though text follows human-defined rules and images can be seen as raw signals of our environment, images and small texts have similar properties. Characters combine to form n-grams, stems, words, phrase, sentences etc. These similar properties make the comparison between computer vision and natural language processing very profitable. In this paper, the focus is on the use of very deep convolutional neural networks for sentence classification tasks. Applying similar ideas to other sequence processing tasks, in particular neural machine translation is left for future research. It needs to be investigated whether these also benefit from having deeper convolutional encoders.