Hashtag Recommendation using Attention-based Convolutional Neural Network

Bekirov Arthur, Dobrenkii Anton
University Innopolis

Introduction

Today microblogging services are used my millions of people, who use them for making short posts. The users also can include hashtags in their text. A hashtag is a string with a hash symbol in front of the words, commonly used for expressing the main idea of the text. With an enormous increase in microblogging services usage among social nets hashtags attract more attention as efficient tools for data mining such as public opinion analysis, microblog retrieval, prediction, etc.
The problem is, that there is a small number of posts with hashtags made by the authors. Thus and automated hashtag recomendation became an interesting area for research.

Background

Due to attention to this problem a big variety of methods were suggested. We can roughly divide them in collaborative filtering, generative and classification models. However, most of them are state-of-the-art solutions those are efficient on a specific task they were made for. According to (Yuyun Gong, 2016) we can use Convolutional Neural Network instead to increase the performance for hashtag recommendation. However, a standard CNN cannot operate with text, so authors propose an attention model for data integration.
Before replicating (Yuyun Gong, 2016) experiment, we tested authors model on images as a solution for wrinkle detection task. There we discovered that the model can be enhanced with Gradient Boosting and, as a result, showing even better performance.

Materials

In the following work we used (Li, Wang, Deng, Wang, & Chang, 2012) dataset containing 50M user tweets as a test set. For training set based on word2vecapi a pretrained twitter dataset with 27B tweets (https://github.com/3Top/word2vec-api) is used.

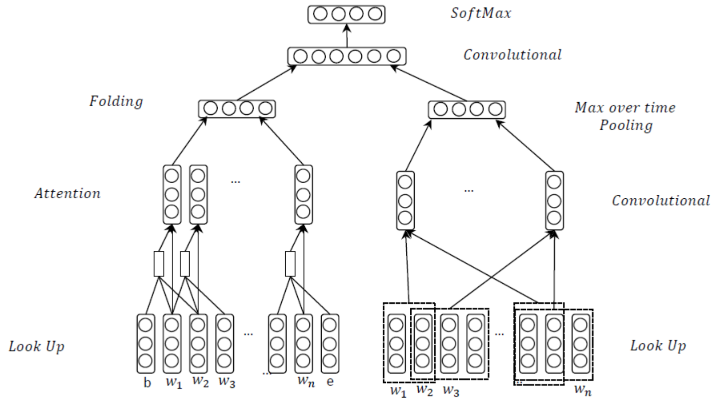


Figure 1. Initial attention-based Convolutional Neural Network

Methods

In this section our solution will be described in details. Consisting of the following parts:

1. Enhancement scheme
2. Data preprocessing
3. Gradient boosting
4. The results

1. Enhancement scheme

Before we start describing our own solution, let's take a look on the model suggested in (Yuyun Gong, 2016). Global and local channels combined are processed through the Softmax function.

As the CNN processes run efficiently, we see that the result can be improved by additional data treatment before we begin analyzing it (dropout - (Nitish Srivastava, 2014)) and after layers combination (Gradient Boosted Trees).

2. Data preprocessing

Our initial dataset from (Li et al., 2012) contained text, tweet, retweet count, URLs, mentioned entities. Thus it consisted of too much information which was odd for our research. What is more, not all the tweets had hashtags made

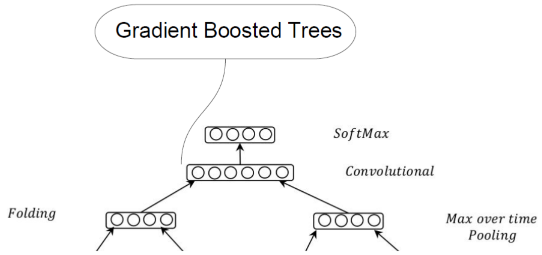
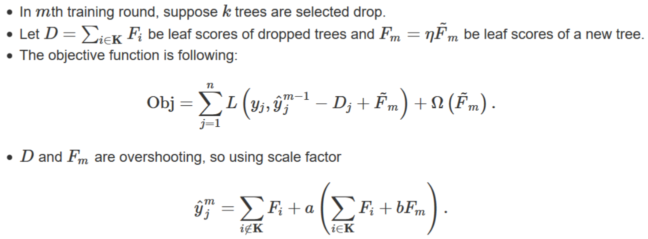


Figure 2. Architecture enhancement



- In mth training round, suppose k trees are selected drop.
- Let $D = \sum_{i \in \mathbf{K}} F_i$ be leaf scores of dropped trees and $F_m = \eta \tilde{F}_m$ be leaf scores of a new tree.
- The objective function is following:

$$\text{Obj} = \sum_{j=1}^{n} L\left(y_j, \hat{y}_j^{m-1} - D_j + \tilde{F}_m\right) + \Omega\left(\tilde{F}_m\right).$$

- D and F_m are overshooting, so using scale factor

$$\hat{y}_j^m = \sum_{i \notin \mathbf{K}} F_i + a\left(\sum_{i \in \mathbf{K}} F_i + bF_m\right).$$

Figure 3. DART XGBoost algorithm

by authors. So firstly we got rid of the odd data and collected 11 279 920 posts with hashtags.

However, the resulting dataset still cannot be processed by CNN. Our next step was text translation into vectors. As we had the pretrained twitter dataset we only had to look for the word in vocabulary. The resulting dataset consisted of strings with vectors and hashtags for final evaluation.

Although CNN is easier to train as it has fewer parameters, we still can effectively use dropout method suggested in (Nitish Srivastava, 2014). The key idea is to randomly drop units (along with their connections) from the neural network during training. This prevents units from co-adapting too much, minimizes overfitting and gives major improvements over other regularization methods.

3. Gradient boosting

As the authors run the results of combined layers only through Softmax function, we suggest to apply GB-Trees with the help of DART booster (K. V. Rashmi, 2015) As DART performs k-fold cross-validation (we take k = 5) we can estimate how accurately a predictive model will perform in practice.

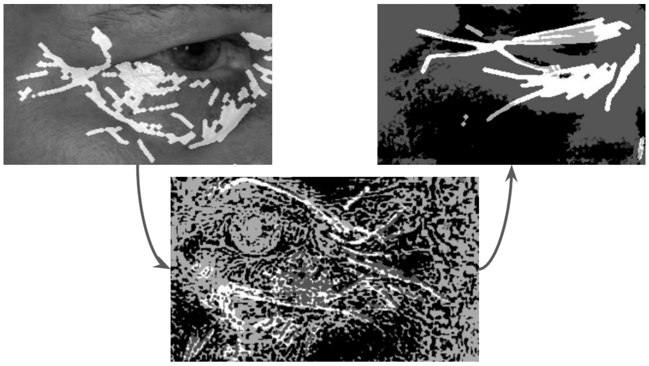


Figure 4. Wrinkle detection task

4. The results

Upon applying the suggested method to images as a solution for wrinkle detection task we managed to outperform the authors model by 4%. Model by (Yuyun Gong, 2016) showed F_1 score = 0.2 while our had F_1 score = 0.208 The inefficiency of scores itself can be explained as F_1 score can't be used properly for images, however we still managed to show better performance.

As for the text results, suggested architecture showed better prediction accuracy, however due to massive size of test set and vocabulary we did not collect the evaluation results before the report submission. We are currently working on the collecting the scores and can update the outcome in the nearest future.

References

K. V. Rashmi, R. G.-B. (2015). Dart: Dropouts meet multiple additive regression trees. , 489-496.

Li, R., Wang, S., Deng, H., Wang, R., & Chang, K. C.-C. (2012). Towards social user profiling: unified and discriminative influence model for inferring home locations. , 1023-1031.

Nitish Srivastava, A. K. I. S. R. S., Geoffrey Hinton. (2014). Dropout: A simple way to prevent neural networks from overfitting. , 929-1958.

Yuyun Gong, Q. Z. (2016). Hashtag recommendation using attention-based convolutional neural network.