Journal of Computer and Communications, 2018, 6, 216-226
http://www.scirp.org/journal/jcc
ISSN Online: 2327-5227
ISSN Print: 2327-5219

Falcon: A Novel Chinese Short Text Classification Method

Haiming Li, Haining Huang, Xiang Cao, Jingu Qian

School of Computer Science and Technology, Shanghai University of Electric Power, Shanghai, China
Email: lhm@shiep.edu.cn, luckydaoge@163.com, caoxiang@mail.shiep.edu.cn, qjg@mail.shiep.edu.cn

How to cite this paper: Li, H.M., Huang, H.N., Cao, X. and Qian, J.G. (2018) Falcon: A Novel Chinese Short Text Classification Method. *Journal of Computer and Communications*, **6**, 216-226.
https://doi.org/10.4236/jcc.2018.611021

Received: September 15, 2018
Accepted: November 24, 2018
Published: November 27, 2018

Abstract

For natural language processing problems, the short text classification is still a research hot topic, with obviously problem in the features sparse, high-dimensional text data and feature representation. In order to express text directly, a simple but new variation which employs one-hot with low-dimension was proposed. In this paper, a Densenet-based model was proposed to short text classification. Furthermore, the feature diversity and reuse were implemented by the concat and average shuffle operation between Resnet and Densenet for enlarging short text feature selection. Finally, some benchmarks were introduced to evaluate the Falcon. From our experimental results, the Falcon method obtained significant improvements in the state-of-art models on most of them in all respects, especially in the first experiment of error rate. To sum up, the Falcon is an efficient and economical model, whilst requiring less computation to achieve high performance.

Keywords

Short Text Classification, Word Vector Representation, One-Hot, Densenet Networks, Convolutional Neural Networks

1. Introduction

Nowadays, short text classification is the task of automatically assigning pre-defined categories to documents written in natural languages. Several types of text categorization have been studied, each of which deals with different types of documents and categories. With the popularity of social media, short text classification becomes an essential component in many applications, such as topic categorization to detect discussed topics, information filtering, and sentiment classification to determine the sentiment typically in product or movie reviews [1].

Compared with the feature of general Chinese short text, the electric power complaint text has the following special characteristics:

- The text relates to the field of electric power, which contains a large number of electrical professional vocabularies.
- The characteristics of the text are not obvious, and more margin information is needed to analyze.
- The text is mixed with too many symbols and numbers.

However, when dealing with shorter text messages, traditional techniques will not perform as well as they would have performed on larger texts.

Some obstructions are encountered when we plan to use Densenet method to deal with electric power complaint text:

- ***Sparse features***: The method of machine learning is to classify or predict based on features, and various features are constructed in text analysis to match the corresponding features. The effect of text classification is depending on features.
- ***Difficult features representation***: The sentence modeling aims at representing sentences as meaningful features of tasks such as text classification. The shorter and simpler for text, the harder feature representation.

In order to solve the above problems, we proposed the Falcon approach, which incorporates Resnet Networks and Densely Connected Convolutional Networks. To the best of our knowledge, no Densenet-based work on short text classification has been proposed to date. Meanwhile, we made several innovations in the data processing to accelerate the model training [2] [3] [4] [5]. The main contributions to our work are as follows:

- ***Feature extraction***: We make innovations on the incorporation model to meet the needs of the classification in short text and make feature extraction become easy.
- ***Reduce the dimension***: In this paper, we use the one-hot vector, PCA dimensionality reduction, vocabulary dictionary and matching the vector id, to avoid additional calculation. The implementation of concat operation will inevitably increase the time complexity of the model.
- ***Marginal feature flow***: We first time propose the new architecture that incorporates Densenet with Resnet in terms of text classification, in order to increase the transfer of margin features.

We apply the proposed model on the short text classification task and achieved superior performance on various benchmarks.

The rest of this paper is organized as follows. In Section 2, we discuss some related works about the models of sentence representation and feature flow. Propaedeutics will be review in Section 3. The Falcon is presented in Section 4. Section 5 carries out the relevant experiment and analyzes the performance of Falcon. Finally, we conclude the paper and make an acknowledgement.

2. Related Work

In this section we first give an overview of the current learning model in feature representation. Next, we review Densenet and channel shuffle that form the basis for Densenet-based Networks.

Models in feature representation. In many recent works of sentence representation, neural network models were constructed on either input word sequences or transformed syntactic parse tree [4]. Among them, Convolutional Neural Network (CNN) gets noticeable achievements. It all started with Kim Yoon [3] that adopted CNN for sentence classification in a simple model architecture (also called TextCNN). Its text matrix is convolved by multiple filters with varying window sizes of for multiple features [6] [7]. Although it can perform well in many NLP tasks, one of its biggest problems is the fixed filter size.

It also has been shown that higher-level modeling on x_l can help to increase the variation in the input, which should then to make it efficient to learn more margin features of between different layers [8]. For example, MSRA's Ho Kaiming team [4] has obtained respectable improvements in deeper neural networks by learning a residual framework. To further improve the information flow between different layers, the densenet was introduced by Dr. Huang Gao [6] direct connections to any layer to all subsequent layers. Xiangyu Zhang *et al.* [9], a new architecture which utilizes pointwise group convolution and channel shuffle, to greatly reduce computation cost while maintaining accuracy.

Another model of mix properties of Text RNN + CNN was put forward by Siwei Lai *et al.* [9]. They apply a recurrent structure to capture contextual information. It obtains semantic vectors by convolving the context vector which is composed of Word, left-side context and right-side context. A disadvantage is that a long training time was consumed. Ying Wen *et al.* [7] improved the model of Siwei Lai [10] by adding a highway layer.

Dense connection. Densely connected networks proposed by Dr. Huang Gao [5] [6] consist of multiple dense blocks, each of which consists of multiple layers. Each layer produces k features, where (K) is referred to as the growth rate of the network. It requires fewer parameters than traditional convolutional networks, as there is no need to relearn redundant feature-maps. Besides better parameter efficiency, another big advantage of Densenets is their improved flow of information and gradients throughout the network, which makes them easy to train.

Channel Shuffle. Xiangyu Zhang *et al.* [11] utilized two new operations, pointwise group convolution and channel shuffle in the architecture of CNN. However, although it achieves significant efficiency improvement for classification on accuracy, its efficiency improvement is less favorable for higher classification accuracy.

In summary, none of them can solve the problems we encountered by analyzing the above model. Hence, we proposed the approach to the next section to address these challenges. We adopted a variant of Densenet to increase the margin feature flow and reducing network complexity. Subsequently, it will achieve great performance on short text classification.

3. Preliminary

Before the model has been proposed, we will review the three state-of-art application of convolutional neural networks to text data.

Consider a word vector group with vocabulary V that is passed through a convolutional network. The network comprises L layers, each layer implement a non-linear transformation $H_l(.)$, where l indexes the layer $H_l(.)$ can be a composite function of operations, which in our case is the "Rectified Linear Unit" (ReLU). We denote the output of the l^{th} layer as x_l.

Resnet. Traditional convolutional feed-forward networks connect the output the l^{th} as input to the $l+1^{th}$ layer: $x_l = H_l(x_{l-1})$. ResNets [11] added a skip-connection that bypassed the non-linear transformations with an identity function:

$$x_l = H_l(x_{l-1}) + x_{l-1} \tag{1}$$

On the one hand, Resnet bypass signal from one layer to the next via identity connections.

On the other hand, Resnet [3] [7] makes this information preservation explicit about additive identity transformations.

Resnet aims to solve the problem of long distance transmission of feature combination problems or shallow information at different levels.

Densenet. To further improve the information flow between different layers, the densenets was introduced direct connections to any layer to all subsequent layers, the layer receives the feature-maps from all of the preceding layers:

$$x_l = H_l([x_0, x_1, \ldots, x_{l-1}]) \tag{2}$$

where x_k, $k \in [0, l-1]$, refers to the concatenation of the feature-maps. In order to implement easily, Densenet concatenates all of the front inputs into a single map.

Crucially, in contrast to Resnet, Densenet never combine features through summation before they are passed into a layer, instead, it combine features by concatenating them. Hence, the l^{th} layer has l inputs, consisting of the feature-maps of all preceding convolutional blocks. Its own feature-maps are passed on to all $L-l$ subsequent layers. This introduces $L(L+1)/2$ connections with a L-layer network, instead of just L, as in traditional architectures.

4. Our Proposed Model

Compressed dimension. Since the dimensionality of region vectors determines the dimensionality of weight vectors, having high-dimensional region vectors means more parameters to learn. If $p|V|$ is too large, the model becomes too complex (w.r.t. the amount of training data available) and training becomes unaffordable expensive even with efficient handling of sparse data; Therefore, one has to lower the dimensionality by lowering the vocabulary size $|V|$ and the region size p, which may or may not be desirable, depending on the nature of the task.

With this representation, we have fewer parameters to learn. Essentially, the expressiveness of our alternative (which loses word order only within small regions) is somewhere between one-hot representation and word2vec.

Concat. One can simply implement a concat operation by adding one more concat layer upon existing networks after the 1×1 convolutional layer (as shown in Equation (3)). The concat operation is following:

$$C_k = R_k \oplus D_k \tag{3}$$

$$x_l = H_l\left(\left[x_0, x_1, \ldots, x_{l-1}\right]\right) \tag{4}$$

where R_k and D_k denote the extracted information at *k-th* step from Resnet and Densenet, $\oplus$ denotes the concat operator.

Average Channel Shuffle. The basic idea of average shuffle algorithm is the index i scans and copies the original data from front to back, between [0, 1] random a index j, the main effect is equivalent to exchang the values of i and j in the copy data, which advantage is that the time consumption is minimal. Which is enabling cross-group information flows for group convolution layers. The pseudocode of Average Shuffle is shown in the following Algorithm 1.

The model schematic architecture, which is shown in Figure 1, different colors represent different features, concat and average shuffle operations are used to the categories of feature map and the flow of global feature information.

Algorithm 1: Average Shuffle

```
1) for each $i \in [1, 9]$ do
2) To shuffle an array $a$ of $n$ elements (indices 0 … $n - 1$);
3) for $i$ from $n - 1$ down to 1 do
4) $j \leftarrow$ random integer with $0 \le j \le i$
5) exchange a [$j$] and a [$i$]
```

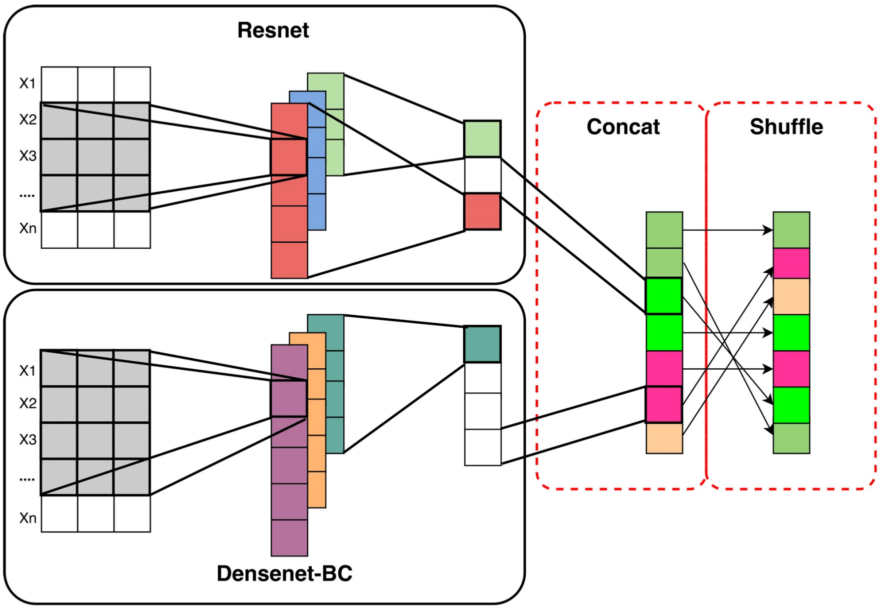


Figure 1. Schematic diagram of model.

5. Experiments and Evaluation

In this section, the effectiveness of the proposed model was validated by THUCNews datasets, and then handle the EPCT datasets. Hardware environment: *4 GB RAM, Nvidia Geforce GTX* 970 *M,* 3 *G*. Software enviroment: All experiments are conducted on a Windows 7 professional 64 bit OS with a simple integrated experimental environment (*anaconda* 3 (64 *bit*) + *python* (3.6) + *spyder*) and an experimental framework of *tensorflow* (1.1.0).

5.1. Datasets and Data Preprocessing

The summary statistics of EPCT datasets are in Table 1.

We divided the dataset into training sets, validation sets, and test sets by 8:1:1. That is, 80% data were used for training the word2vec and classifier. In constructing the word vector model, the size of word vectors was set to 50 (*i.e.*, each word was represented as a 50-dimensional vector.). In word vector representation, each word is represented as a vector in an arbitrary vector space. Then every word is represented as a numerical vector, we can compute relevancy between words. Continuous word vectors representation techniques have been proposed in [12] [13] [14]. The proposed two models such as "continuous bag of words" and "continuous skip-gram" can express an aspect of meaning of words. Both models are implemented to word2vec [15] [16] [17]. The word2vec is a tool which realizes word vector representations to text set. The whole preprocessing workflow is shown in Figure 2.

Table 1. Description of datasets.

Date sets	Classes	Number	Ave Length
THUNews	20	20,000	264
EPCT	7	5000	93

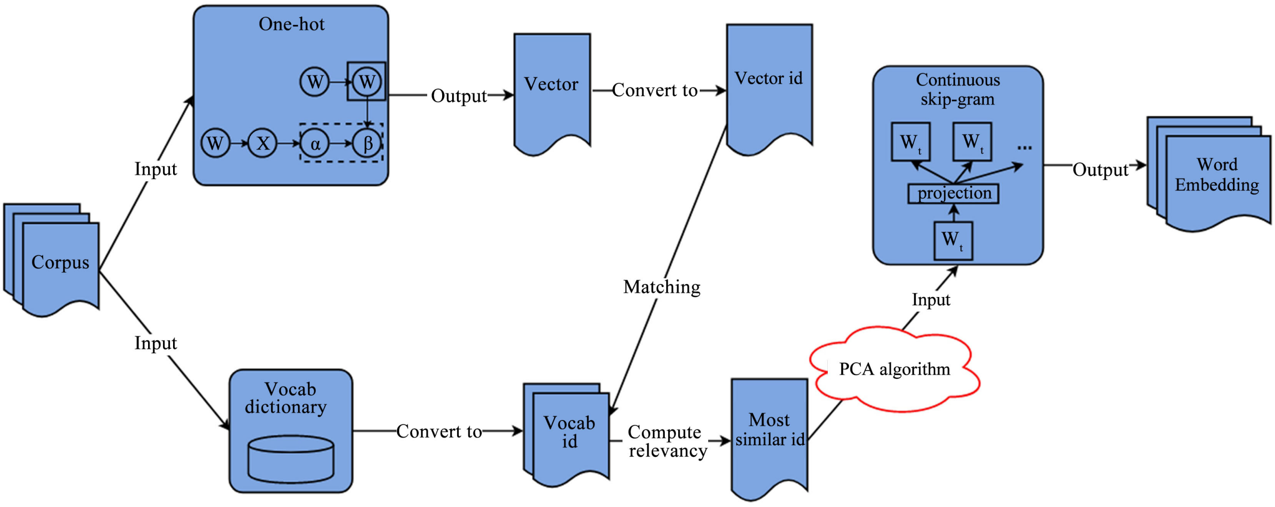


Figure 2. Workflow of data preprocessing.

5.2. Performance Results

The concat operation used in Equation (3) is not valid when the size of feature-maps changes. For convolutional layers with filter size 3 × 3 (group convolution) and 1 × 1, The transition layers used in our experiments consist of a batch normalization layer and an 1 × 1 convolutional layer followed by a 2 × 2 average pooling layer.

Several parameters of our model are summarized in Table 2.

5.2.1. One-Hot Vs. Word2vec

Table 3 shows the error rates of our proposed model in comparison with the baseline methods. The first thing to note is that on all the datasets, our model outperforms the baseline methods, which demonstrates the effectiveness of our approach.

To look into the details, on this task, while our model outperform all the baseline methods, which indicates that in this setting the merit of having fewer parameters is larger than the benefit of keeping word order in each region.

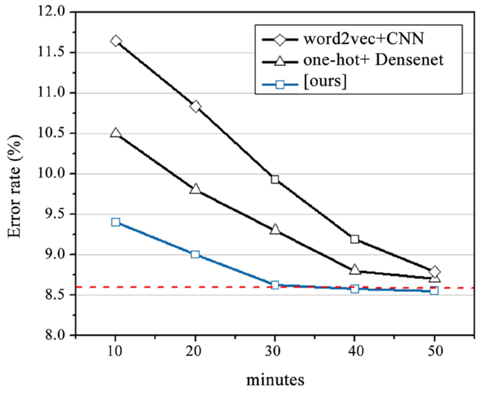
5.2.2. Concat Vs. No Concat on EPCT

We first perform a set of experiments to validate the F1-score on Resnet, Densenet-BC and Concat model. Table 4 results that beyond most competing methods are bold. All the results of are obtained using Concat operation. It's obvious that Concat model performs better by a large margin, especially in "suggestion" and "complaint".

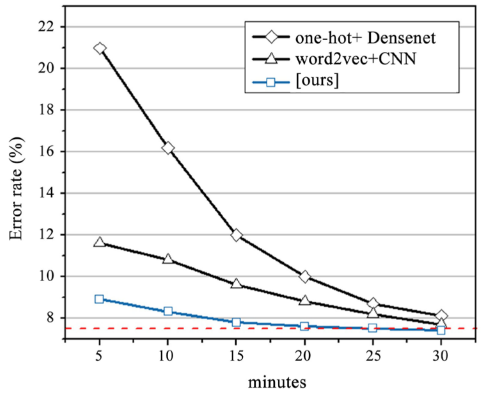
Finally, the results with training sets of various sizes on THUNews and EPCT are shown in Figure 3.

5.2.3. Comparison with State-of-the-Art Models

On THUNews and EPCT datasets, the best error rates we obtained by training were 8.6 and 7.5. Without exception, which is both better than other methods. Meanwhile, we also find that our model will achieve good performance in a short training time, which is shown in Figure 4. Since excellent performances were reported on short text classification, we presume that their model is optimized for short sentences, but not for text categorization in general.

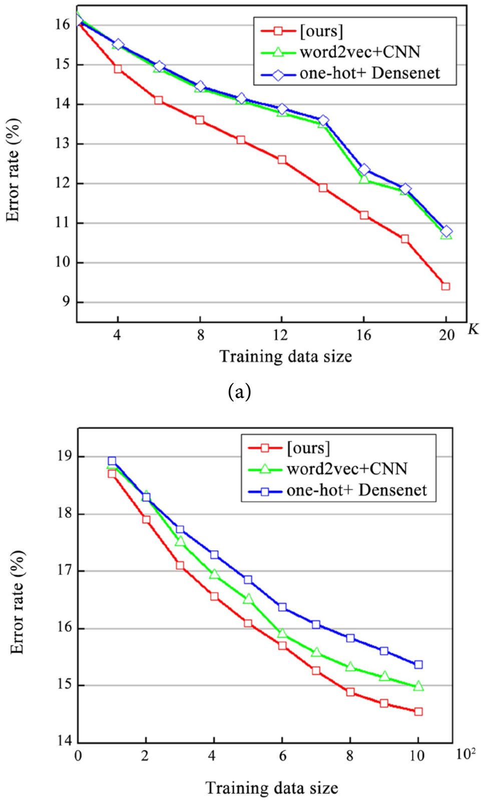


(a)



(b)

Figure 3. Comparisons of error rate on THUNews (a) and EPCT (b).



(a)

(b)

Figure 4. Error rate in relation to training data size.

Table 2. Parameters in our model.

Description	Values
embedding dim	64
seq length	600
vocab size	500
hidden dim	128
batch size	64
num epochs	10

Table 3. Error rate (%) comparison with baseline methods. Short text classification on THUNews (2 K training documents) and EPCT (0.5 K training documents) indicates that most frequent word vector were used.

Methods	THUNews	EPCT
one-hot + CNN	11.47	9.50
word2vec + CNN	8.46	8.21
one-hot + Densenet	8.34	7.92
word2vec + Densenet	8.21	7.75
[ours]	**8.06**	**7.63**

Table 4. Model with/without concat (select the persuasive benchmark F1-score as an example.).

Categories	Model		
—	Densenet	Resnet	Concat model
praise	0.76	0.60	0.76
fault repair	0.91	0.86	**0.93**
suggestion	0.83	0.75	**0.89**
report	0.76	0.62	0.76
complaint	0.75	0.69	**0.89**
business consultation	0.96	0.88	**0.97**
opinion	0.65	0.59	**0.72**

6. Conclusions

In this paper, this is the first time that a novel short text classification method based on Densenet networks was proposed to address the electric power complaint text. Meanwhile the improvements in short text classification model making a comprehensive comparison of the classification efficiency were implemented. The model provides a new approach and a train of thought for the study of the other short Chinese texts.

In addition, the highly consistent classification results were found by comparison of the actual classification results of the state grid; this is due to the improvement of feature learning method in this model, just like reading, the richer the content of the book, the more knowledge you will learn.

Although some improvements have been achieved, one of the future works is to develop a more efficient short text classification model in enlarging short text features.

Acknowledgements

To acknowledge all of the people that have contributed to this paper, the donors of THUNews dataset and the anonymous reviewers for their helpful feedback and suggestions. This work is also supported by the National Natural Science

Foundation of China (No. 61272437 and 71203137).

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Aggarwal, C.C. and Zhai, C.X. (2012) A Survey of Text Classification Algorithms. Mining Text Data. Springer US, 163-222. https://doi.org/10.1007/978-1-4614-3223-4_6

[2] Cho, K., Van Merrienboer, B., Gulcehre, C., *et al.* (2014) Learning Phrase Representations Using RNN Encoder-Decoder for Statistical Machine Translation. arXiv:1406.1078 [cs.CL]

[3] He, K., Zhang, X., Ren, S., *et al.* (2015) Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. arXiv:1502.01852 [cs.CV]

[4] Huang, G., Liu, S., Laurens, V.D.M., *et al.* (2017) CondenseNet: An Efficient DenseNet Using Learned Group Convolutions. arXiv:1711.09224 [cs.CV]

[5] Huang, G., Liu, Z., Maaten, L.V.D., *et al.* (2017) Densely Connected Convolutional Networks. *IEEE Conference on Computer Vision and Pattern Recognition*, **1**, 2261-2269. https://doi.org/10.1109/CVPR.2017.243

[6] Ioffe, S. and Szegedy, C. (2015) Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *Proceedings of the* 32*nd International Conference on Machine Learning*, *PMLR*, **37**, 448-456.

[7] Iyyer, M., Enns, P., Boyd-Graber, J., *et al.* (2014) Political Ideology Detection Using Recursive Neural Networks. *Proceedings of the* 52*nd Annual Meeting of the Association for Computational Linguistics*, **1**, 1113-1122. https://doi.org/10.3115/v1/P14-1105

[8] Kalchbrenner, N., Grefenstette, E. and Blunsom, P. (2014) A Convolutional Neural Network for Modelling Sentences. arXiv:1404.2188 [cs.CL]

[9] Krizhevsky, A., Sutskever, I. and Hinton, G.E. (2012) ImageNet Classification with Deep Convolutional Neural Networks. *International Conference on Neural Information Processing Systems*, Curran Associates Inc., 1097-1105.

[10] Chen, G., Ye, D., Xing, Z., *et al.* (2017) Ensemble Application of Convolutional and Recurrent Neural Networks for Multi-Label Text Categorization. 2017 *International Joint Conference on Neural Networks* (*IJCNN*), Anchorage, 14-19 May 2017, 2377-2383. https://doi.org/10.1109/IJCNN.2017.7966144

[11] Liao, Q. and Poggio, T. (2016) Bridging the Gaps between Residual Learning, Recurrent Neural Networks and Visual Cortex.

[12] Mikolov, T., Chen, K., Corrado, G., *et al.* (2013) Efficient Estimation of Word Representations in Vector Space.

[13] Mikolov, T., Sutskever, I., Chen, K., *et al.* (2013) Distributed Representations of Words and Phrases and Their Compositionality. *International Conference on Neural Information Processing Systems*, Daegu, 3-7 November 2013, 3111-3119.

[14] Mikolov, T., Yih, W.T. and Zweig, G. (2013) Linguistic Regularities in Continuous Space Word Representations.

[15] Ji, Y.L. and Dernoncourt, F. (2016) Sequential Short-Text Classification with Re-

current and Convolutional Neural Networks. 515-520.

[16] Zhang, X., Zhou, X., Lin, M., *et al.* (2017) ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices.

[17] Zhang, Y. and Wallace, B. (2015) A Sensitivity Analysis of (and Practitioners' Guide to) Convolutional Neural Networks for Sentence Classification.