A New Short Text Sentimental Classification Method Based on Multi-Mixed Convolutional Neural Network

Hao Lidong
School of Computer Science and Software Engineering
East China Normal University Shanghai, China
Shanghai, China
e-mail: haolidongecnu@163.com

Zhao Hui*
Shanghai Key Laboratory of Trustworthy Computing
Shanghai, China
School of Computer Science and Software Engineering
Shanghai, China
e-mail: hzhao@sei.ecnu.edu.cn

Abstract—**Almost all e-commerce platforms provide online product comments service. The comments are some words of mouth about the products. Customers usually reference the comments to make an informal decision when buying similar products. In addition, companies would like to know the feedbacks about the products through the comments. However, the typical big data characteristic and noisy short text data characteristic make the online comment data analysis becoming a challenge work. In this work, we propose a *Multi-Mixed Convolutional Neural Network (MMCNN)* model to analyze the sentiment of online product comments. We mix the convolution and pooling features in mixed layer to enhance effectiveness of the online comments sentimental analysis. The skip-gram model is used to train the word vector. Because the length of each comment is not fixed, two new empirical matrix filling methods are designed which are cyclic matrix filling and random matrix filling. We apply our approach for two datasets which are online comments about infant power crawled from www.jd.com and online reviews about hotel crawled from www.elong.com. Experiment results demonstrate the effectiveness of our approach in comparison with Support Vector Machine, Maximum Entropy, Naive Bayesian and classic CNN.**

Keywords-feature fusion; convolution neural network; matrix filling; sentiment classification

I. INTRODUCTION

With the flourishing development of online shopping mall, the product comments service provided by the e-commerce platform have become one of the channels to evaluate the product quality and one of the references for the customers to purchase products. In addition, companies are more interested in the positive or negative results of the comments to understand their own product quality and to formulate marketing strategies. Different from the classic long text data processing, the online product comment is usually a short text with more oral words, has limitation of contextual data, contains a lot of noise due to the use of incorrect grammar. Therefore, how to analyze the emotion of product comments data effectively becomes a challenge work for researchers and draws extensive attention from researchers. Traditional dictionary based methods mainly depend on the effectiveness and quality of the constructed dictionary to some extent. The dictionary is commonly established using WordNet [1] or HowNet [2]. Most online comments sentimental analysis methods use a method from the field of machine learning, such as SVM [3], Naive Bayes [4], Conditional Random Fields [5], Maximum Entropy [6] which are based on feature selection. The feature selection and combination play an important role for detecting the sentiment of a comment. Deep learning method uses neural networks to learn many levels of abstraction that extracts features from basic levels to advanced levels step by step to establish a mapping from basic signal to high-level semantics which is similar to human brain structure [7].

In this work, we propose a Multi-Mixed Convolutional Neural Network (MMCNN) that exploits word level information to analyze the sentiment of online product comments. In MMCNN, features are extracted in convolutional layer and pooling layer and the extracted features are combined in the fully-connected layers. To solve the problem of different length of comments, two new empirical matrix filing methods are designed. We perform experiments that show the effectiveness of MMCNN on two comment datasets. One is online comments about infant powder sold on www.jd.com, a famous online shopping mall in China. Another one is the reviews about hotel on www.elong.com,a famous online hotel reservation platform. Additionally, in our experiments we provide the contribution of cyclic matrix filling and random matrix filling to fix the problem of different size of comment.

This paper is organized as follows. In Section 2, we discuss some related work. In Section 3, we describe the proposed Multi-Mixed Convolutional Neural Network. In section 4, the empirical matrix filling methods are presented. Section 5 details our experimental setup and results. Finally, in Section 6 we conclude our works.

II. RELATED RESEARCH WORKS

Traditional textual sentiment analysis that worked well with well-written long text faces challenges when analyzing short noisy online comments data. The traditional dictionary based sentiment classification methods depends on the efficiency and quality of the constructed dictionary. WordNet [1] or HowNet [2] are commonly used dictionaries. The common used statistic learning methods are SVM [3], Naive Bayes [4], Conditional Random Fields [5], Maximum Entropy [6]. Zhang et al. [3] used word2vec to represent reviews text and SVM is applied to classify the sentiments. Manek et al. [8] used Gini coefficients for feature selection and SVM is also used to analyze the sentiment of movie reviews. Compared with the dictionary-based method, the statistic learning method reduces the dependence on sentiment dictionaries. However, the feature selection greatly affects the final classification accuracy.

Compared with the traditional sentiment analysis method, deep learning based methods have drawn the researcher extensive attention in solving the problem of large-scale textual sentiment analysis. He Yanxiang et al. built the emotional feature-space matrix by using emoticons in Microblog and combined it with convolutional neural network, which effectively enhanced the accuracy of emotional classification of Microblog comment data [9]. Some researchers dedicated to optimize the convolutional neural network based model to improve the performance, such as changing the architecture of network, improving the activation functions and nesting network in network.

Soujanya Poria et al. presented a novel way of extracting features from short texts, based on the activation values of an inner layer of a deep convolutional neural network [10]. He K et al. proposed a method to solve the disappearance of the gradient caused by the over-deep network through the residual network [11]. The most notable non-saturated activation functions are used in Rectified Linear Unit (ReLU) [12]. It is a piecewise linear function which prunes the negative part to zero, and retains the positive part. In contrast to ReLU, Maas et al. proposed leaky ReLU in which the negative part is totally dropped and leaky ReLU is assigned a noon-zero slope [13]. Randomized Leaky Rectified Linear unit (RRelu) is the randomized version of leaky ReLU [14] which first be used in Kaggle NDSB Competition. He et al. proposed parametric rectified linear unit to deal with the large scale image classification which has better accuracy than that of ReLu [15]. Lin M et al. proposed a novel deep network structure called Network in Network (NIN) to enhance the model discriminability for local patches within the receptive field [16].

The above researches mainly focused on enhancing the performance of convolutional neural network by changing the structure of the network, improving the activation functions and nesting network in the network. Our work puts the efforts on the network structure optimization. We propose a novel CNN called Multi-Mixed Convolution Neural Network (MMCNN) for the online comments sentimental analysis.

III. MULTI-MIXED CONVOLUTION NEURAL NETWORK MODEL

A. *Convolutional Neural Network*

Convolutional Neural Network (CNN or ConvNet) is a class of deep, feed-forward artificial neural networks that has successfully been applied to analyzing visual imagery. CNNs use a variation of multilayer perceptrons designed to require minimal preprocessing [17]. CNN consists of one input layer, multiple hidden layers which are convolutional layers, pooling layers, fully connected layers, and one output layer. Convolutional layers apply a convolution operation to the input, passing the result to the next layer. The convolution emulates the response of an individual neuron to visual stimuli. Pooling layers combine the outputs of neuron clusters at one layer into a single neuron in the next layer. Convolution leverages sparse interactions parameter sharing, and representations to help improve a machine learning system. Moreover, convolution provides a means for working with inputs of variable. Figure 2 shows an example of sentiment classification based on CNN.

B. *Multi-Mixed Convolutional Neural Network*

The CNN has successfully been applied to analyzing visual imagery. Online comments are short texts and always have character limitation. The word embedding produced by unsupervised pre-training using Word-Level Embedding representation is high-dimensional and sparse. Unlike image RGB information, adjacent point in word vector has not strong correlation. In addition, the classic convolutional neural network only uses convolution layers to extract features which makes it impossible to observe the feature information from multi-view. We proposed a Multi-Mixed Convolutional Neural Network that exploits word level information to perform sentiment analysis of online product comments. The network extracts feature from the word-level. The word vector matrix is taken as the input, and then is not only passed to convolution layers but also to pooling layers. Features extracted from convolution layers and pooling layers are mixed in the mixed layer which we proposed in CNN. The main contribution in our network architecture is the mixed layer on which the multiple convolutional vectors and multiple pooling vectors are mixed. The architecture of MMCNN is presented in details below.

The first layer of the network is the input layer which transforms words into real-valued word vector. Given a comment consisting of m words, all words are converted into a matrix $\mathrm{M_j} = \{\mathrm{x_1}; \mathrm{x_2}; \dots; \mathrm{x_m}\}$ as the input vector of the MMCNN where x_i is the *ith* word vector.

Hidden layer is composed of multiple convolution layers and pooling layers in which features are extracted and combined. Given a group of convolutional filters $k^m = \{k_1^m, k_2^m, \dots, k_n^m\}$ whose size is $m \times n$. The convolutional features set is $xc^m = \{xc_1^m, xc_2^m, \dots, xc_n^m\}$, where

$$xc_j^m = f\left(\sum_{i \in M_j} k_j^m \times x_j + b_j^m\right) \quad (1)$$

The convolution-pooling feature set is $xcp^m = \{xcp_1^m, xcp_2^m, \dots, xcp_n^m\}$, where

$$xcp_j^m = f\left(down\left(xc_j^m\right) \times \beta_j^m + bp_j^m\right) \quad (2)$$

The max polling or average pooling is used to get xcp^m. The matrix $M_j = \{x_1; x_2; \dots; x_m\}$ is input into the pooling layer, thus the pooling feature $xp = \{xp_1, xp_2, \dots, xp_n\}$

$$xp_j = f(down(x_j) \times \beta_j + p_j) \quad (3)$$

where b is convolutional bias, bp is the pooling bias after convolutional calculation and p is the pooling bias, *f (·)* is the activation function, *down (·)* is the down sampling function.

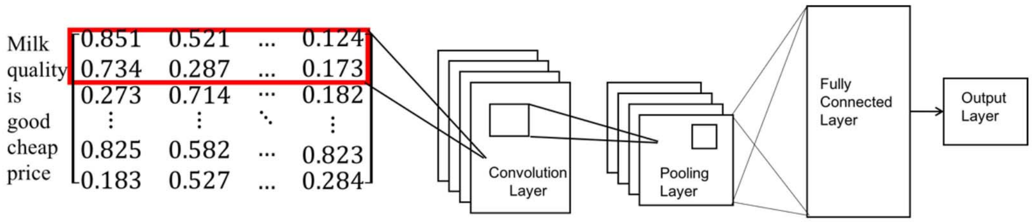

Figure 1. An Example of Sentiment Analysis based on CNN

Mixed layer is in hidden layers. The convolutional features, convolution-pooling features and pooling features are mixed in this layer. Because the row vector and column vector of features are in different size, the features vectors are needed to transform into row vectors firstly and then the features are mixed. Here we give the definitions about the mixed operation.

Define 1: $reshape(\cdot)$ is a function which converts a m×n matrix into a row vector V_l.

$$V_l = \mathrm{reshape}(M_{mn}) = \mathrm{reshape}\left(\begin{bmatrix} a_{11} & \cdots & a_{1n} \\ \cdots & \cdots & \cdots \\ a_{m1} & \cdots & a_{mn} \end{bmatrix}\right) = [a_{11} \quad \cdots \quad a_{1n} \quad \cdots \quad a_{mn}] \tag{4}$$

Define 2: Operation ⊗ concatenates row vector V_p and row vector V_q to a row vector V_c.

$$V_l = V_p \otimes V_p = [a_1 \quad \cdots \quad a_n] \otimes [b_1 \quad \cdots \quad b_m] = [a_1 \quad \cdots \quad a_n \quad \cdots \quad b_m] \tag{5}$$

Define 3: Operation $\sum_{\otimes}$ concatenates a series of vectors to a row vector:

$$\sum\nolimits_{\otimes} V = V_1 \otimes V_2 \otimes \cdots \otimes V_n \tag{6}$$

Thus, the feature vector which consists of convolutional features, convolution-pooling features and pooling features is represented as the follow.

$$W = \sum\nolimits_{\otimes} reshape(xc_j^m) \otimes \sum\nolimits_{\otimes} reshape(xcp_j^m) \otimes \sum\nolimits_{\otimes} reshape(xcp_j^m) \tag{7}$$

This feature vector W is input to fully connected layer to compute the possibility of sentimental tendency using formula (8) and formula (9). The *softmax* classifier is applied. The possibility that the comments is in negative sentiment is:

$$\mathrm{P}(\mathrm{result} = \mathrm{neg}) = \frac{1}{1 + e^{-W^T x}} \tag{8}$$

The possibility that the comments is in positive sentiment is:

$$P(result = pos) = \frac{e^{-W^T x}}{1 + e^{-W^T x}} \tag{9}$$

IV. Two Empirical Matrix Filling Methods

Technically, zero vector is used to fill into the word embedding matrix because of the different comments' length. This practical method leads to no features extracted in convolutional layer. To avoid this problem, we propose two new matrix filling methods which are cycle matrix filling method and random matrix filling method. Given an $m \times n$ matrix, the basic idea of cycle matrix filling method is that the word vector of a comment is filled to a row vector of the matrix repeatedly until the norm of the row vector is n. The basic idea of random matrix filling method is that all the words vector of the comment is filled to a row vector of the matrix only once and then a word vector is selected randomly to fill to the row vector until the norm of the new vector is n.

Given a comment sentence consisting of L words $Sen = \{w_0, w_1, \ldots, w_L\}$. The word vector set generated based on the Skip-gram algorithm is represented as Rep_N. N is the norm of the word vector. Let $maxlen$ be the word number of the longest comment after the word segmentation. $M = \{M_1, .. M_n\}$ stands for the word embedding matrix. Two algorithms are presented below.

Algorithm 1. Cycle Matrix Filling

Function $cycleMatrixFilling(\cdot)$
Input: $Sen = \{w_0, w_1, \ldots, w_l\}$ //the words of a comment
Rep_N //the word vector set
Output: Matrix M

```
initial Maxtix M = [ ]
j = 0
while i ≤ maxlen do:
    i = 0
    for each w_i in Sen do:
      if w_i can be found in Rep_N do:
        put the value of Rep_N[w_i] to M_j
      else:
        put zero vector to M_j
      end if
    i = i + 1
    j = j + 1
    end for
end while
```

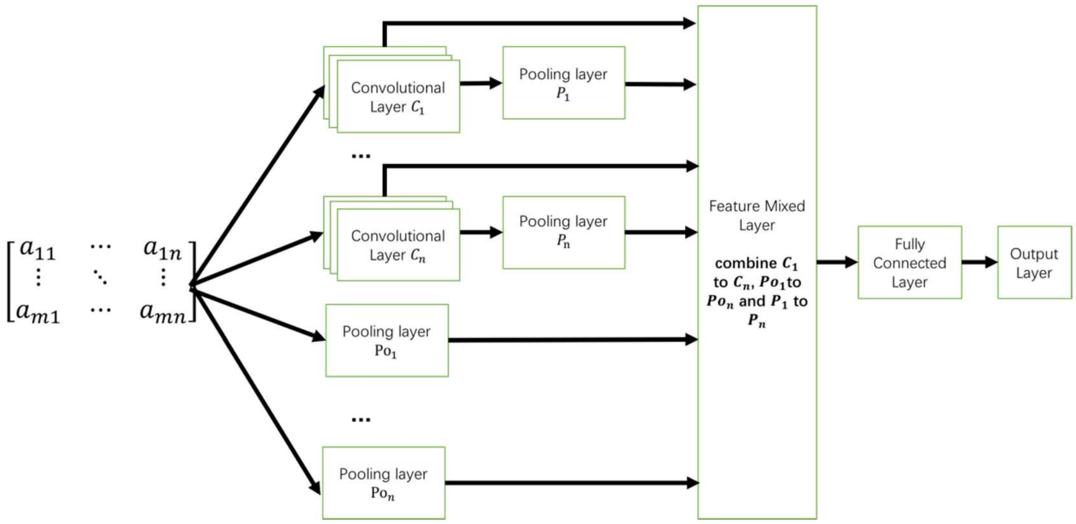

Figure 2. The structure of Multi-Mixed Convolutional Neural Network

Algorithm2. Random Matrix Filling

Function $randomMatrixFilling(\cdot)$

Input: Sen $= \{\mathrm{w}_0, \mathrm{w}_1, \dots, \mathrm{w}_l\}$

//the words of a comment

$\mathrm{Rep_N}$ //the word vector set

Output: Matrix M

initial Maxtix M = [ ]
j = 0
for each $\mathrm{w_i}$ in Sen do
  i = 0
  if $\mathrm{w_i}$ can be found in $\mathrm{Rep_N}$ do
    put the value of $\mathrm{Rep_N[w_i]}$ to $\mathrm{M_j}$
  else
    put zero vector to $\mathrm{M_j}$
    end if
  i = i + 1
  j = j + 1
end for
while i ≤ maxlen do:
  select a w randomly in Sen do
    if w can be found in $\mathrm{Rep_N}$ do:
      put the value of $\mathrm{Rep_N[w_i]}$ to $\mathrm{M_j}$
    else:
      put zero vector to $\mathrm{M_j}$
    end if
  j = j + 1
  end for
end while

Take the online comments for infant milk powder for example, the matrix without using our methods is shown below.

$$\begin{bmatrix} -0.8585 & 1.9576 & -0.1238 & \cdots & 0.6382 & -1.0330 \\ 0.4197 & 0.09170 & -0.4496 & \cdots & 1.1996 & 0.0088 \\ 0.54920 & -0.8735 & -0.3956 & \dots & 0.5289 & 0.0372 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0.0000 & 0.0000 & 0.0000 & \cdots & 0.0000 & 0.0000 \\ 0.0000 & 0.0000 & 0.0000 & \cdots & 0.0000 & 0.0000 \end{bmatrix}$$

The matrix using cycle matrix filling is shown below.

$$\begin{bmatrix} -0.8585 & 1.9576 & -0.1238 & \cdots & 0.6382 & -1.0330 \\ 0.4197 & 0.0917 & -0.4496 & \cdots & 1.1996 & 0.0088 \\ 0.5492 & -0.8735 & -0.3956 & \dots & 0.5289 & 0.0372 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ -0.8585 & 1.9576 & -0.1238 & \cdots & 0.6382 & 0.6382 \\ 0.4197 & 0.0917 & -0.4496 & \cdots & 1.1996 & 0.0088 \end{bmatrix}$$

The matrix using *random matrix filling* is shown below.

$$\begin{bmatrix} -0.8585 & 1.9576 & -0.1238 & \cdots & 0.6382 & -1.0330 \\ 0.4197 & 0.0917 & -0.4496 & \cdots & 1.1996 & 0.0088 \\ 0.5492 & -0.8735 & -0.3956 & \dots & 0.5289 & 0.0372 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0.5492 & -0.8735 & -0.3956 & \cdots & 0.5289 & 0.0372 \\ 0.4197 & 0.0917 & -0.4496 & \cdots & 1.1996 & 0.0088 \end{bmatrix}$$

V. EXPERIMENTS

A. *Technical Roadmap of the Our Work*

The technical roadmap of online product comments sentimental analysis based on MMCNN is shown in Figure 3. Firstly, the word vectors are trained using Chinese Corpus of Wikipedia. Secondly, representation of comments is generated based on the trained word vectors and the matrix filling method is applied to solve the different length problem of comments. Then, the MMCNN model is trained.

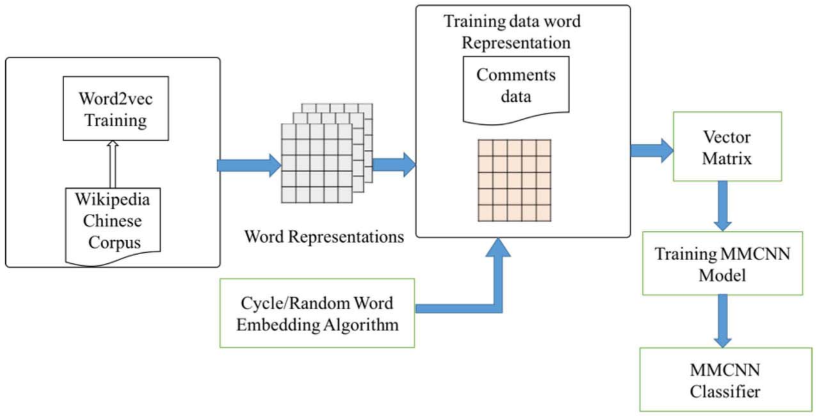

Figure 3. The Technical roadmap of online product comments Sentimental Analysis based on MMCNN

B. *Experiment Preparation*

We investigate the empirical performances of our proposed MMCNN models on two different datasets.

Dataset1: The dataset is the online comments about infant milk power sold in Jingdong Mall (www.jd.com). 46,000 comments generated from 2015 to 2017 are crawled. We labelled the comments with 4 stars or 5 stars as positive comments. The comments with 1 star and 2 stars are labelled as negative comments. The comments with 3 stars are ignored. There are 23000 positive comments and 23000 negative comments.

Dataset2: The dataset is the online hotel reviews generated from 2016 to 2017 on YiLong (www.elong.com). 12,000 comments are crawled. “Need Improvement” regards as negative comments. “Recommend” regards as positive comments. There are 6000 positive comments and 6000 negative comments.

The Chinese Wikipedia corpora is used to train the word vector. The experiment is conducted on the computer which is configured with the 8GB memory, 256GB SSD, CPU i7 7700HK, GPU GTX1080Ti (memory 11GB) and windows10 64-bit operating system. Tensorflow1.2 is employed as the deep learning framework.

To evaluate the performance of our model, we adopt the Accuracy metric, which is defined as:

$$\mathrm{Acc} = \frac{\mathrm{T}}{\mathrm{N}} \tag{10}$$

where T is the number of correctly predicted samples, N is the total number of samples. Accuracy measures the percentage of correct predicted sample in all samples.

C. *Hyperparameters Settings*

In our experiments, we compare the Accuracy of MMCNN model based method with SVM, Naive Bayes, Maximum Entropy Model and classical convolutional neural network based methods. *Chi-Square* feature selection is applied in SVM, Naive Bayes and Maximum Entropy methods. Classical convolutional neural network’s *minibatch* is set to 100 and dropout is set to 0.5. 3×3 convolution layer is employed. 1×1, 3×3, 5×5 convolutional layer and 2×2 max-pooling layer are used in MMCNN. Other parameters are set as that of CNN.

D. *Experiment Results*

Figure 4 and Figure 5 show the Accuracy of MMCNN model based method, classical convolutional neural network based methods and SVM, Naive Bayes, Maximum Entropy Model based method on Dataset1 and Dataset2 respectively.

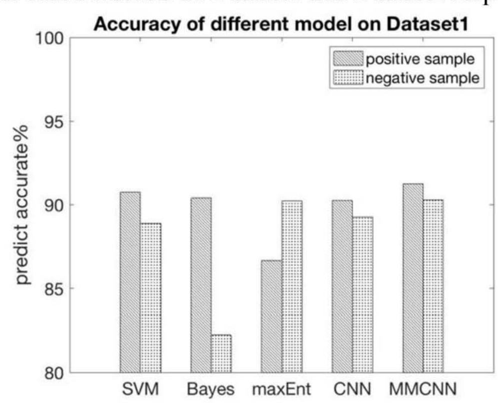

Figure 4. The Accuracy of different model on Dataset1

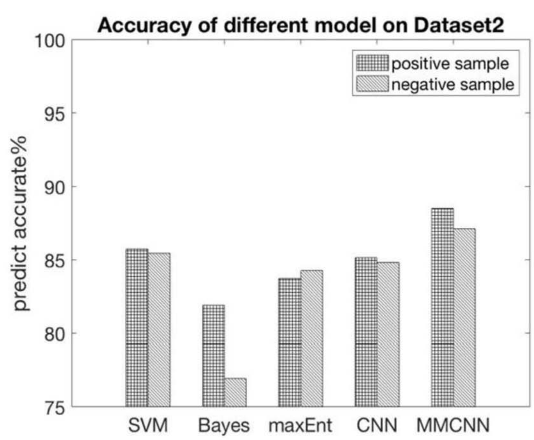

Figure 5. The Accuracy of different model on Datase2

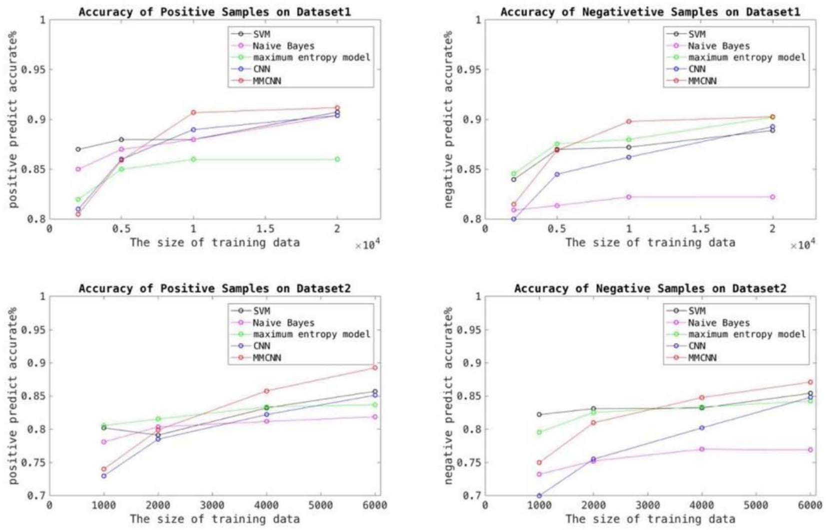

Figure 6. The Accuracy of Different Size Negative Samples

As Figure 6 shows, with the volume of sample ascending, the accuracy of SVM, Naive Bayes, Maximum Entropy Model and classical convolutional neural network based methods has no improvement when the number of comment record is over 10000. However, both the accuracy of Convolutional Neural Networks-based approach and MMCNN-based approach increase.

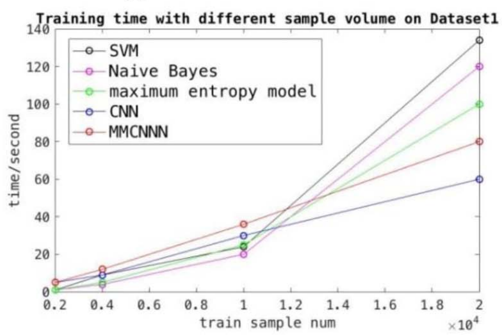

Figure 7. Training time with different sample size on Dataset1

We also compare the model training time using diffident size dataset. Figure 7 shows that as the volume of data increases, all the training time of SVM, Naive Bayes and Maximum Entropy are exponential growth. While the training time of CNN and MMCNN model based method is basically a linear growth.

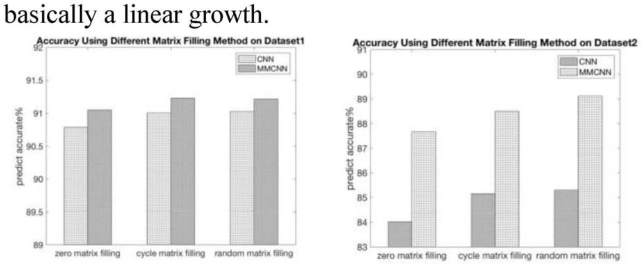

Figure 8. Accuracy Using Different Matrix Filling Method

The Accuracy employing two matrix filling methods are computed to evaluate the effectiveness. Figure 8 displays the accuracy of CNN and MMCNN model based method using cyclic matrix filling method, random matrix filling method with the traditional method respectively. The results show the matrix filling method we proposed are effective.

VI. CONCLUSIONS

The online comments about products sold by the online shopping mall have become the electronic word of mouth. They are draw extensive attention only from customers but also from companies. However, the online comments are usually a short text with more oral words and have limitation of contextual data. They always contain a lot of noise due to the use of incorrect grammar. Thus, effective sentimental analysis of the online comments is a challenge work compared with that of the classic long text data processing. In this paper, we propose a *Multi-Mixed Convolutional Neural Network* for online comments sentimental analysing. The MMCNN focuses on generating more features and concatenating the convolutional layer features, convolution-pooling features and pooling features to a mixed layer. MMCNN is benefit from the feature fusion which is always ignored in other models. To solve the sparse problem of the word embedding matrix, we proposed two new matrix filling methods. Experiments on online comments about infant milk powder and online reviews about hotel indicate the effectiveness of our work.

ACKNOWLEDGMENT

This work was supported by grants from National Key R&D Program of China (2016YFC1300300).

REFERENCES

[1] Fellbaum C. WordNet[M]//Theory and applications of ontology: computer applications. Springer Netherlands, 2010: 231-243.

[2] Xianghua F, Guo L, Yanyan G, et al. Multi-aspect sentiment analysis for Chinese online social reviews based on topic modeling and HowNet lexicon[J]. Knowledge-Based Systems, 2013, 37: 186-195.

[3] Zhang D, Xu H, Su Z, et al. Chinese comments sentiment classification based on word2vec and SVM perf[J]. Expert Systems with Applications, 2015, 42(4): 1857-1863.

[4] Gamallo P, Garcia M. Citius: A Naive-Bayes Strategy for Sentiment Analysis on English Tweets[C]//SemEval@ COLING. 2014: 171-175.

[5] Marcheggiani D, Täckström O, Esuli A, et al. Hierarchical Multi-label Conditional Random Fields for Aspect-Oriented Opinion Mining[C]//ECIR. 2014: 273-285.

[6] Gautam G, Yadav D. Sentiment analysis of twitter data using machine learning approaches and semantic analysis[C]//Contemporary computing (IC3), 2014 seventh international conference on. IEEE, 2014: 437-442.

[7] LeCun Y, Bengio Y, Hinton G. Deep learning[J]. Nature, 2015, 521(7553): 436-444.

[8] Manek A S, Shenoy P D, Mohan M C, et al. Aspect term extraction for sentiment analysis in large movie reviews using Gini Index feature selection method and SVM classifier[J]. World wide web, 2017, 20(2): 135-154.

[9] HE Yan-Xiang, SUN Song-Tao, NIU Fei-Fei et al. A Deep Learning Model Enhanced with Emotion Semantics for Microblog Sentimental Analysis [J]. CHINESE JOURNAL OF COMPUTERS, 2017, 40(4): 773-790. (in Chinese)

[10] Poria S, Cambria E, Gelbukh A. Deep convolutional neural network textual features and multiple kernel learning for utterance-level multimodal sentiment analysis[C]//Proceedings of the 2015 conference on empirical methods in natural language processing. 2015: 2539-2544.

[11] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 770-778.

[12] Hinton G, Deng L, Yu D, et al. Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups[J]. IEEE Signal Processing Magazine, 2012, 29(6): 82-97.

[13] Maas A L, Hannun A Y, Ng A Y. Rectifier nonlinearities improve neural network acoustic models[C]//Proc. ICML. 2013, 30(1)

[14] Srivastava N, Hinton G E, Krizhevsky A, et al. Dropout: a simple way to prevent neural networks from overfitting[J]. Journal of machine learning research, 2014, 15(1): 1929-1958.

[15] He K, Zhang X, Ren S, et al. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification[C]//Proceedings of the IEEE international conference on computer vision. 2015: 1026-1034.

[16] Lin M, Chen Q, Yan S. Network in network[J]. arXiv preprint arXiv:1312.4400, 2013.

[17] LeCun Y. LeNet-5, convolutional neural networks[J]. URL: http://yann. lecun. com/exdb/lenet, 2015.