Paper 12 | Recurrent Convolutional Neural Networks for Text Classification by Siwei Lai et. al – published in 2015

Abstract

1. Traditional text classifiers often rely on many human-designed features, such as dictionaries, knowledge bases and special tree kernels
2. This paper introduces a recurrent CNN for text classification without human-designed features. The model has a recurrent structure to capture contextual information as far as possible when learning word representations
3. Employ a max-pooling layer that automatically judges which words play key roles in text classification to capture the key components in texts
4. **The experimental results show that the proposed method outperforms the state-of-the-art methods on several datasets, particularly on document-level datasets**

Introduction

1. A key problem in text classification is feature representation, which is commonly based on BoW model, whereby n-grams patterns are extracted as features
2. There are also other feature selection methods:
    - Frequency
    - Mutual information (MI)
    - Latent semantic analysis (pLSA)
    - Latent Dirichlet Allocation (LDA)
3. Traditional methods often ignore contextual information or word order in texts and although high-order n-grams can fix this issue, it still experiences data sparsity problem (the problem of not observing enough data in a corpus to model the language accurately)
4. Richard Socher proposed the **Recursive NN** that has been proven to be efficient in terms of constructing sentence representations. However, the model captures semantics via a tree structure which means that performance heavily depends on the performance of the textual tree construction.
    - Constructing such a textual tree exhibits a time complexity of at least O(n^2), where n is the length of the text.
    - Relationship between two sentences can hardly be represented by a tree structure, meaning that Recursive NN is unsuitable for modelling long sentences or documents
5. **Recurrent Neural Network** only exhibits a time complexity of O(n), whereby the model analyses a text word by word and stores the semantics of all the previous text in a fixed-sized hidden layer
    - This model better captures contextual information and it works with long texts
    - However, the model is bias, whereby the later words are more dominant than earlier words
6. **CNN** is used to tackle this bias problem. It has a time complexity of O(n). However, when using convolutional kernels, it is difficult to determine the window size. Small window sizes may result in the loss of information whereas large windows result in an enormous parameter space

7. To address the limitation of all the models above, the paper proposed **Recurrent Convolutional Neural Network (RCNN)**
    - First, we apply a bi-directional recurrent structure to capture the context information when learning word representations (which may introduce a considerably less noise compared to a traditional window-based NN)
    - Second, we employ a max-pooling layer that identify which features play key roles in text classification
    - By combining the recurrent structure and max-pooling layer, we utilise the advantage of both RNN and CNN
    - The proposed model has time complexity of O(n)

Models

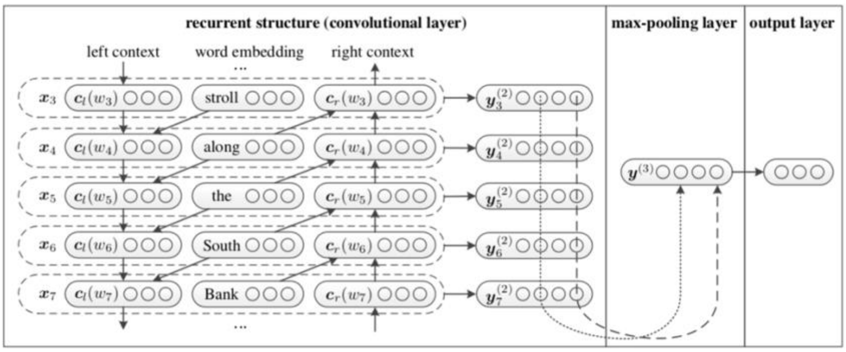


Figure 1: The structure of the recurrent convolutional neural network. This figure is a partial example of the sentence "A sunset stroll along the South Bank affords an array of stunning vantage points", and the subscript denotes the position of the corresponding word in the original sentence.

1. Word representation learning
    - Combine a word and its context to present a word whereby the context should give us a more precise word meaning
    - Use a bi-directional recurrent NN to capture the contexts
    - The context vector captures the semantics of all left- and right-side contexts
        - For example: "A sunset stroll along the South Bank affords an array of stunning vantage points"
        - $C_l(w_7)$ – encodes the semantics of the left-side context of the word "Bank" – "A sunset stroll along the South"
        - $C_r(w_7)$ – encodes the semantics of the right-side context of the word "Bank" – "affords an array of stunning vantage points"
    - Finally, we define the representation of word w_i as the concatenation of the left-side context vector, the word embedding, and the right-side context vector. This way, the model may be better able to disambiguate the meaning of the word w_i

      $$\boldsymbol{x}_i = [\boldsymbol{c}_l(w_i); \boldsymbol{e}(w_i); \boldsymbol{c}_r(w_i)]$$

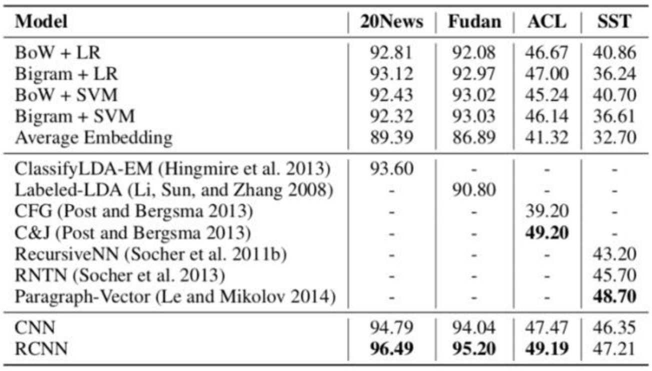
    - We apply a linear transformation together with tanh activation function to x_i and send result to next layer

      $$\boldsymbol{y}_i^{(2)} = \tanh\left(W^{(2)}\boldsymbol{x}_i + \boldsymbol{b}^{(2)}\right)$$

    - where the y is a **latent semantic vector**, in which each semantic factor will be analysed to determine the most useful factor for representing the text

2. Text representation learning
    - The pooling layer converts texts with various lengths into a fixed-length vector
    - The max-pooling layer attempts to find the most important latent semantic factors in the document
3. Training
    - SGD to optimise training target. In each step, we randomly select an example and make a gradient step. Initialise all the parameters from a uniform distribution
    - The paper uses the Skip-gram model to pre-train the word embedding

Results



Model	20News	Fudan	ACL	SST
BoW + LR	92.81	92.08	46.67	40.86
Bigram + LR	93.12	92.97	47.00	36.24
BoW + SVM	92.43	93.02	45.24	40.70
Bigram + SVM	92.32	93.03	46.14	36.61
Average Embedding	89.39	86.89	41.32	32.70
ClassifyLDA-EM (Hingmire et al. 2013)	93.60	-	-	-
Labeled-LDA (Li, Sun, and Zhang 2008)	-	90.80	-	-
CFG (Post and Bergsma 2013)	-	-	39.20	-
C&J (Post and Bergsma 2013)	-	-	**49.20**	-
RecursiveNN (Socher et al. 2011b)	-	-	-	43.20
RNTN (Socher et al. 2013)	-	-	-	45.70
Paragraph-Vector (Le and Mikolov 2014)	-	-	-	**48.70**
CNN	94.79	94.04	47.47	46.35
RCNN	**96.49**	**95.20**	**49.19**	47.21

Table 2: Test set results for the datasets. The top, middle, and bottom parts are the baselines, the state-of-the-art results and the results of our model, respectively. The state-of-the-art results are reported by the corresponding essays.

1. Comparison between proposed model and traditional text classification method and the state-of-the-art approaches
2. The results show that the neural network approaches tend to outperform the traditional methods, which process that NN can effectively compose the semantic representation of texts
3. CNNs and RCNNs vs RecursiveNNs (SST dataset)
    - Convolutional-based approaches achieve better results, showing that it is more suitable for constructing the semantic representation of texts. We believe that the main reason for this is that CNN can select more discriminative features through max-pooling layer and capture contextual information through convolutional layer
4. The proposed model outperforms state-of-the-art methods in three of the 4 datasets
5. RCNN vs CNN
    - This might be due to the recurrent structure capturing contextual information better than the window-based structure in CNNs
    - The paper investigated further the ability of the recurrent structure to capture contextual information
        - o The authors consider all odd window sizes from 1 – 19 to train and text the CNN model
        - o In the figure below, we observe that the RCNN outperforms the CNN for all window sizes, showing that the recurrent structure can capture contextual information without relying on window size. This is because the recurrent structure can preserve longer contextual information and introduces less noise

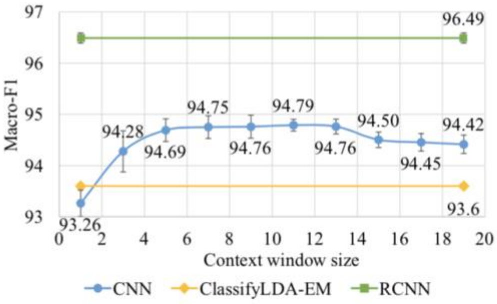
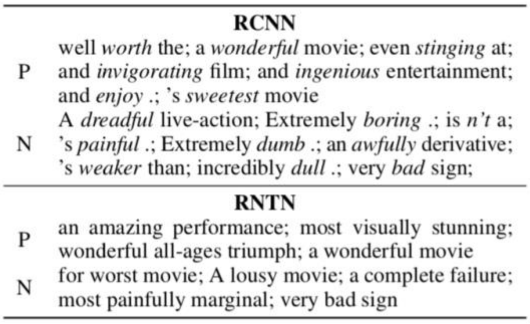


Figure 2: Macro-F1 curve for how context window size influences the performance of the 20Newsgroups classification

- Learned keywords



	RCNN
P	well *worth* the; a *wonderful* movie; even *stinging* at; and *invigorating* film; and *ingenious* entertainment; and *enjoy* .; 's *sweetest* movie
N	A *dreadful* live-action; Extremely *boring* .; is *n't* a; 's *painful* .; Extremely *dumb* .; an *awfully* derivative; 's *weaker* than; incredibly *dull* .; very *bad* sign;
	RNTN
P	an amazing performance; most visually stunning; wonderful all-ages triumph; a wonderful movie
N	for worst movie; A lousy movie; a complete failure; most painfully marginal; very bad sign

Table 3: Comparison of positive and negative features extracted by the RCNN and the RNTN

  - The result demonstrate that the most important words for positive sentiment are words such as “worth”, “sweetest”, and “wonderful” and those for negative sentiment are words such as “awfully”, “bad”, and “boring”

Conclusion

1. The model captures contextual information with the recurrent structure and constructs the representation of text using a CNN. The results show that the proposed model outperforms CNN and RecursiveNN using four different text classification datasets