Extractive Document Summarization Based on Hierarchical GRU

Yong Zhang, Jinzhi Liao, Jiuyang Tang, Weidong Xiao, and Yuheng Wang
National University of Defense Technology, Changsha, Hunan, 410072, China
495289761@qq.com

Abstract*—Neural network has provided an efficient approach for extractive document summarization, which means selecting sentences from the text to form the summary. However, there are two shortcomings about the conventional methods: they directly extract summary from the whole document which contains huge redundancy, and they neglect relations between abstraction and the document. The paper proposes TSERNN, a two-stage structure, the first of which is a key-sentence extraction, followed by the Recurrent Neural Network-based model to handle the extractive summarization of documents. In the extraction phase, it conceives a hybrid sentence similarity measure by combining sentence vector and Levenshtein distance, and integrates it into graph model to extract key sentences. In the second phase, it constructs GRU as basic blocks, and put the representation of entire document based on LDA as a feature to support summarization. Finally, the model is tested on CNN/Daily Mail corpus, and experimental results verify the accuracy and validity of the proposed method.

Keywords-Document summarization; Two-stage; RNN; LDA

I. INTRODUCTION

Document summarization is a significant problem in natural language processing. A summary is the main objective of summarization method, which can be categorized into extraction and abstraction. Extraction concentrates on selecting specific sentences from the corpus and chronological concatenation to construct a summary, whereas abstraction generates novel sentences from information which sometimes is a part of the corpus for long text or all of short text.

Essentially, document summaraization is a human cognitive problem. Based on the understanding of whole paper, someone is able to form a summary. With the booming of machine learing, new structure such as neural network is proposed to simulate human behavior. Thus, document summarization has got a huge leap.

Based on neural network, a variety of methods have been proposed to handle extractive summarization. For instance, on the task of multi-document extractive summarization Yin [28] utilized Convolutional Neural Networks (CNN) to project sentences to continuous vector space and then select sentences by minimizing the cost based on their prestige and diverseness. CNN was also utilized by Cao [5] to solve the problem of query-focused multi-document summarization. The paper applied weighted-sum pooling over sentence representations to represent documents, where weights are learned from attention over sentence representations based on the query.

With the emergence of strong generative neural models for text [2], abstractive techniques are also becoming increasingly popular. For example, an attentional feed-forward network was proposed by Rush [24] for abstractive summarization of sentences into short headlines. Nallapati [22] developed Rush's work, and put forward a set of recurrent neural network based encoder-decoder models, which focus on various aspects of summarization like handling out-of-vocabulary words and modeling syntactic features of words in the sentence.

Generally speaking, abstractive summarization is closer to the human way of thinking. After understanding documents, the model reconstructs and simplifies them into a summary. Recently, however, Abigail See [25] proves the sophisticated abstractive model performs worse than straightforward extraction, which only takes first three sentences into account. Cheng [7] proposed an attentional encoder-decoder for extractive single-document summarization and applied it to the CNN/Daily Mail corpus. Therefore, we propose a two-stage extractive recurrent neural network classifier (TSERNN) summarization model of single documents using neural networks.

Contributions. To summarize, the key contributions of the paper is threefold:

1. This paper utilizes two-stage approach method to simplify long texts into several sentences, and then put these sentences into RNN. The boost of computing efficiency is prominent.
2. This paper experimentally applied the representation of the entire document, based on latent dirichlet allocation (LDA), to support summarization.
3. The proposed method outperforms other methods in the same corpus, and it is demonstrated to enjoy significant advantage in both terms of efficiency and accuracy.

Organization. After introduction in Section 1, Section 2 surveys related works on document summarization. Necessary background knowledge, and the proposed TSERNN methods are introduced in Section 3. Experimental studies are reported in Section 4, followed by conclusion given in Section 5.

II. RELATED WORKS

With more attention drawn into extractive summarization, there have been a large number of methods that were proposed to cope with the problem.

Prior work, namely modeling sentence similarity as well as optimizing selection process, occupies a large proportion.

978-1-5386-6580-0/18/$31.00 ©2018 IEEE
DOI 10.1109/ICRIS.2018.00092

The pioneering work LexRank [12] and LexPageRank [11] both computed cosine similarity based on TF-IDF (Term Frequency-Inverse Document Frequency) matrix first, then used ranking algorithm PageRank to calculate sentence prestige. In addition,HITS, another typical ranking algorithm in web mining [16] is also popularly studied in document summarization [27]. In [1], a method was presented to measure dissimilarity between sentences using the normalized google distance [9], then then sentence clustering is performed to select the most distinctive sentences from each cluster to form summaries. Yin [29] combined longest common sub-sequence (LCS) [6], weighted LCS, skip-bigram statistic with word semantic similarity derived by Latent Dirichlet Allocation [4] for sentence similarity learning, and exploited traditional PageRank to select sentences.

In order to improve coherence of generated summaries, extractive models are supposed to have a discourse structure that is similar to that of the source document. Rhetorical Structure Theory (RST) [19] is one way of introducing the discourse structure of a document to a summarization task in [14]. They proposed a noisy-channel model that used RST. Although their method generated a well-organized summary, no optimality of information coverage was guaranteed and their method could not accept large texts because of the high computational cost. Recently, RST trees is transformed into dependency trees and used for single document summarization in [21]. They formulated the summarization problem as a tree knapsack problem with constraints represented by the dependency trees. There have been some studies that have used discourse structures locally to optimize the order of selected sentences [23].

Different from the methods above, in view of the successful applications of representation learning by deep neural network in lots of natural language processing (NLP) tasks [10,15,18], this paper proposes a novel representation learning approach TSERNN. Utilizing a sequence classifier to tackle document summarization problems has been attempted by Shen [26]. In his work, Conditional Random Fields was used to binary-classify sentences sequentially. The proposed method seems more concise. Because RNNs is applied in the model, handcrafted features, which are used to represent sentences and documents in [26], are not required.

III. PROPOSED METHOD

In this section, the framework of the proposed model is outlined , and the details of the model structures involved in the two-stage are followed thereafter.

A. Model Framework

In proposed method, the extractive summarization problem is transformed into a sequence classification problem. Document is firstly computed through fisrt stage method to extract key sentences, which are still in the original document order, and then a classifier is laid on the top of RNNs model to decide whether or not the worked senctence should be contained in the summary. After computing the original document through the first stage method, the worked sentences are then put into the RNNs. Selected by classifier, the sentences finally form a summarization, and the whole frame is showed in Fig.1.

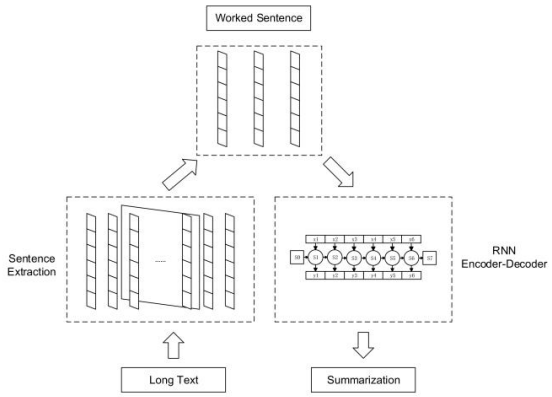

Figure. 1: Overall Framework of Proposed Method

B. First stage: Sentence Extraction

To handle the problem of key sentence extraction, this paper utilizes a graph model, where a hybrid sentence similarity is measured by combining sentence vector similarity and Levenshtein distance. In the model, a document is expressed as a complete graph, in which each node represents a sentence and edges weight the similarity of connecting two nodes, as shown in Fig.2.

Pagerank [17] is then employed to rank the graph, and this paper selects the fixed top-ranked sentences in the original order to form a summary.

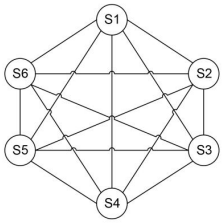

Figure. 2: Structure of Graph Model for Extraction

Similarity Caculation Traditional method uses TF-IDF [11] to estimate sentence importance. For TF-IDF is only a simple calculation of the coincidence degree, it is weak in finding the semantic similarity between words. As the core to determine sentence importance is semantic similarity, this paper proposes a similarity measure that combines Levenshtein distance and sentence vector similarity, so as to capture similarity both semantically and literally.

Levenshtein distance is a string metric for measuring the difference between two sequences. Informally, Levenshtein distance between two words is the minimum number of single-character edits-insertions, deletions or substitutionsrequired to change one word into the other.

Inspired by work on learning vector representations of words using neural networks, this paper utilizes the incorporation of sentence vector, where every sentence is mapped to a unique vector, represented by a column in

matrix. In aspect to words, every word is also mapped to a unique vector, and then get averaged or concatenate with sentence vector to predict the next word in a context. After being trained, the sentence vectors can be used as features for the sentence. Therefore, by calculating the Cosine function of the sentence vectors, the model can obtain the semantic similarity of the two sentences, and the high Cosine-value means high similarity.

The formal definition about the similarity of two sentences Si and Sj is expressed below,

$$(S_i, S_j) = \alpha \cdot \cos(V(S_i), V(S_j)) + (1-\alpha) \cdot Lev(S_i, S_j).$$

V (Si) means the sentence vector of Si, α represents the weight in semantic similarity, Lev(Si, Sj) denotes the Levenshtein distance.

Scoring Strategy Denote G = (V, E) as an undirected-graph with the set of vertices V and set of edges E. For a given vertex Vi, In(Vi) is the set of vertices that point to it (predecessors), and Out(Vi) is the set of vertices that vertex Vi points to (successors). In addition, wij represents the weight of the edge between the Vi and Vj node, where the value of wij is calculated using the similarity sim(Si, Sj). The score of the vertex Vi is defined as follows

$$S(V_i) = (1-d) + d \sum_{V_j \in In(V_i)} \frac{w_{ij}}{\sum_{V_k \in Out(V_j)} w_{jk}} S(V_j),$$

where d is a damping factor between 0 and 1, which has the role of integrating into the model the probability of jumping from a given vertex to another random vertex in the graph. In practice, d is usually set to 0.85.

C. Second stage: Summary Generation

The basic building block of the model is constructed on GRU based Recurrent Neural Network [8]. Specifically speaking, there are two gates, update gate u and reset gate r, of GRU-RNN. New memory is computed by the former memory and the current input, and reset gate would decide importance about former memory to the new one, and update gate would judge whether the former memory should be transmitted to next state. The procedure can be expressed in following equations:

$$u_j = \sigma(W_{ux} x_j + W_{uh} h_{j-1} + b_r),$$
$$r_j = \sigma(W_{rx} x_j + W_{rh} h_{j-1} + b_r),$$
$$h_j^{'} = \tanh(W_{hx} x_j) + W_{hh}(r_j \bullet h_{j-1}) + b_h,$$
$$h_j = (1-u_j) \bullet h_j^{'} + u_j \bullet h_{j-1},$$

where W's are weight matrixes and b's are bias of the GRU-RNN. h_j is the output at timestep j. x is the corresponding input vetor. means the Hadamard product, and the whole process is visualized as in Fig.3.

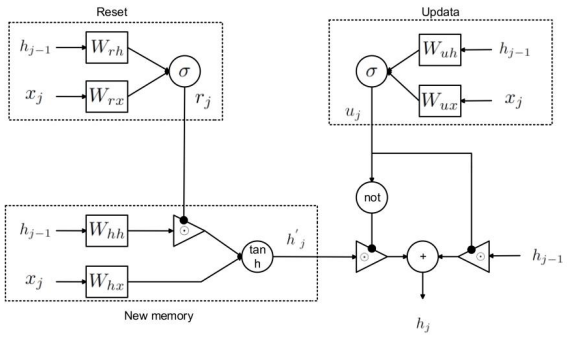

Figure. 3: GRU Struture

A two-layer bi-directional GRU-RNN is the core of the model, where the first layer runs at word level and sencond layer runs at sentence level. In the first layer, hidden state representations is computed at each word position, based on the current word embeddings and the previous hidden state. Then a backward RNN is utilized to run from the last word to the first, and then the forward and backward RNNS form a bidirectional RNN. The bidirectional RNN is also applied in second layer, where it runs at the sentence-level and accepts the concatenated hidden states of the former bi-directional word-level RNNs as input.

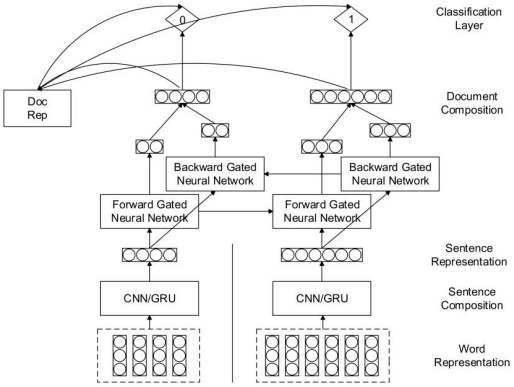

Figure. 4: A two-layer RNN based sequence classifier

Furthermore, Latent Dirichlet Allocation (LDA) [4] is utilized to generate representation d of the entire document, which is regarded as an additional feature to support summarization. LDA is a generative statistical topic model, and provides a powerful framework to collect topics in documents. The idea of the model is that documents are represented as weighted relevancy vectors on latent topics, in which each topic is characterized by a distribution over words based on hierarchical Bayesian models of a corpus [3]. In LDA, each documents can be expressed in several topics with different weights, and the number of topics and the proportion of words are considered as two hidden variables.

For classification, a logistic layer is laid on the top to decide whether the sentence should be contianed in the summary. As shown below,

$$P(y_j = 1 \mid h_j, s_j, d) = \sigma(W_s h_j + h_j^T W_d d - h_j^T W_r \tanh(s_j) + b),$$

yj is a binary variable indicating whether the summary containes jth sentence, hj the output of bidirectional RNN, and sj is the dynamic representation of the summary at the jth sentence position,

$$s_j = \sum_{i=1}^{j-1} h_i (y_j = 1 \mid h_j, s_j, d).$$

Wshj means the jth sentence's content, hjT Wdd denotes the salience of the sentence with respect to the document, hjT Wrtanh(sj) represents the redundancy of the sentence on current state of the summary.

IV. Experimental Evaluation

In this section, the paper reports the experimental studies with in-depth analyses.

A. Experiment Settings

Corpora CNN/DailyMail corpus[13] is utilized, which contains online news articles (781 tokens on average)paired with multi-sentence summaries (3.75 sentences or 56 tokens on average), for experiment. and the paper also adopt the method proposed in [7] for the task of extractive document summarization as contrast test.

In order to make a fair comparison with the former, the paper left out the CNN subset of the corpus, as done by them. To compare with the latter, joint CNN/Daily Mail corpora was used in the test. Overall, there are 196,557 training documents, 12,147 validation documents and 10,396 test documents from the Daily Mail corpus. If CNN subset is included, there are 286,722 training documents, 13,362 validation documents and 11,480 test documents. On average, there are about 28 sentences per document in the training set, and an average of 3-4 sentences in the reference summaries. The average word count per document in the training set is 802.

Criteria Different variants of the Rouge metric are used in this work to evaluate the performance of TSERNN. Lead-3 model, which simply produces the leading three sentences of a document as the summary, is used as a baseline on all datasets. On the Daily Mail corpora, a feature-rich logistic classifier, LReg, is an additional evaluation, which is used as a baseline by Cheng [7].

TSERNN Settings In the phase of sentence extraction, the damping coefficient in PageRank d is set to 0.85, and the number of key sentences extracted is set to 8. When calculating sentence similarity, the semantic level similarity weight is $\alpha = 0.75$.

In the phase of summary generation, the paper used 100-dimensional word2vec [20] embeddings trained on the worked sentences as the embedding initialization. Then the paper fixed the model hidden state size at 200, and used a batch size of 64 at training time, and adadelta [30] to train our model.

Instead of setting a threshold, like P(y = 1) ≥ 0:5, to select the sentences for forming summary, the model picks sentences sorted by the predicted probabilites until exceeding the length limit when limited-length Rouge is used for evaluation.

B. Experiment Result

Table 1

Methods	Rouge-1	Rouge-2	Rouge-L
Lead-3	21.8	7.2	11.7
LReg(500)	18.6	6.9	10.2
Cheng's work	22.5	8.6	12.4
TSERNN-2	25.4	10.1	13.9
TSERNN	**26.5**	**11.4**	**15.1**

Performance on entire Daily Mail test set

Daily Mail corpus As TABLE I shows, it's obvious to notice that TSERNN outperforms other baselines, and more attention should be laid on TSERNN-2 which means the corpora is directly put into RNNs whitout being simplified through stage-one. One reason for the poor performance of TSERNN-2 is that whole original documents contain many redundanct information, which may add more noise to RNNs in generating summary.

Table 2

Methods	Rouge-1	Rouge-2	Rouge-L
Lead-3	39.2	15.6	35.6
Nallapati's work	35.3	13.1	32.2
TSERNN-2	39.1	15.7	34.8
TSERNN	**40.4**	**16.9**	**36.1**

Performance on CNN/Daily Mail test set

CNN/Daily Mail corpus On the joint CNN/Daily Mail corpus, the paper compares the proposed method with the abstractive encoder-decoder based model proposed in [22], for it is the only work that reports performance on this dataset. The outcome, shown in TABLE II, is not unexpected, because abstractive summarization is a much harder problem, which has been proved in [25].

V. Conclusion

In this work, a two-stage structure is propose to solve the extractive summarization problem. With firstly selected key sentences from the document and then put into RNNs which contains document-representation based on LDA, the model outperforms other deep learning models.

In the future work, there are two aspects taken into account to enhance the method. On the one hand, abstractive approaches will be considered to be combined with the model, which means the model may pre-train the model using abstractive training. On the other hand, the

model used in key sentences selection can be replaced by some recent work to boost the accuracy of the model.

REFERENCES

[1] Aliguliyev, R.M.: A new sentence similarity measure and sentence based extractive technique for automatic text summarization. Expert Syst. Appl. 36(4), 7764-7772(2009)

[2] Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. CoRR abs/1409.0473 (2014)

[3] Blei, D.M., Griffiths, T.L., Jordan, M.I., Tenenbaum, J.B.: Hierarchical topic models and the nested chinese restaurant process. In: Advances in Neural Information Processing Systems 16 [Neural Information Processing Systems, NIPS 2003, December 8-13, 2003, Vancouver and Whistler, British Columbia, Canada]. pp. 17-24(2003)

[4] Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. Journal of Machine Learning Research 3, 993-1022 (2003)

[5] Cao, Z., Li, W., Li, S., Wei, F., Li, Y.: Attsum: Joint learning of focusing and summarization with neural attention. In: COLING 2016, 26th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, December 11-16, 2016, Osaka, Japan. pp. 547-556 (2016)

[6] Chali, Y., Joty, S.R.: Unsupervised approach for selecting sentences in query-based summarization. In: Proceedings of the Twenty-First International Florida Artificial Intelligence Research Society Conference, May 15-17, 2008, Coconut Grove, Florida, USA. pp. 47-52 (2008)

[7] Cheng, J., Lapata, M.: Neural summarization by extracting sentences and words. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers(2016)

[8] Chung, J., G¨ul¸cehre, C¸ ., Cho, K., Bengio, Y.: Empirical evaluation of gated recurrent neural networks on sequence modeling. CoRR abs/1412.3555 (2014)

[9] Cilibrasi, R., Vit´anyi, P.M.B.: The google similarity distance. IEEE Trans. Knowl. Data Eng. 19(3), 370-383 (2007)

[10] Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., Kuksa, P.P.: Natural language processing (almost) from scratch. Journal of Machine Learning Research 12, 2493-2537 (2011)

[11] Erkan, G., Radev, D.R.: Lexpagerank: Prestige in multi-document text summarization. In: Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing , EMNLP 2004, A meeting of SIGDAT, a Special Interest Group of the ACL, held in conjunction with ACL 2004, 25-26 July 2004, Barcelona, Spain. pp. 365-371 (2004)

[12] Erkan, G., Radev, D.R.: Lexrank: Graph-based lexical centrality as salience in text summarization. J. Artif. Intell. Res. 22, 457-479 (2004)

[13] Hermann, K.M., Kocisk´y, T., Grefenstette, E., Espeholt, L., Kay, W., Suleyman, M., Blunsom, P.: Teaching machines to read and comprehend. In: Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada. pp. 1693-1701 (2015)

[14] III, H.D., Marcu, D.: A noisy-channel model for document compression. In: Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, July 6-12, 2002, Philadelphia, PA, USA. pp. 449{456 (2002)

[15] Kalchbrenner, N., Grefenstette, E., Blunsom, P.: A convolutional neural network for modelling sentences. In: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, June 22-27, 2014, Baltimore, MD, USA, Volume 1: Long Papers. pp. 655-665 (2014)

[16] Kleinberg, J.M.: Authoritative sources in a hyperlinked environment. J. ACM 46(5), 604-632 (1999)

[17] L. Page, S. Brin, R.M., Winograd, T.: The pagerank citation ranking: Bringing order to the web (1999)

[18] Le, Q.V., Mikolov, T.: Distributed representations of sentences and documents. In: Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014. pp. 1188-1196 (2014)

[19] Mann, W.C., Thompson, S.A.: Assertions from discourse structure. In: Strategic Computing - Natural Language Workshop: Proceedings of a Workshop Held at Marina del Rey, California, USA, May 1-2, 1986 (1986)

[20] Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States. pp. 3111-3119 (2013)

[21] Morita, H., Sasano, R., Takamura, H., Okumura, M.: Subtree extractive summarization via submodular maximization. In: Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics, ACL 2013, 4-9 August 2013, Sofia, Bulgaria, Volume 1: Long Papers. pp. 1023-1032 (2013)

[22] Nallapati, R., Xiang, B., Zhou, B.: Sequence-to-sequence rnns for text summarization. CoRR abs/1602.06023 (2016)

[23] Nishikawa, H., Hasegawa, T., Matsuo, Y., Kikui, G.: Opinion summarization with integer linear programming formulation for sentence extraction and ordering. In: COLING 2010, 23rd International Conference on Computational Linguistics, Posters Volume, 23-27 August 2010, Beijing, China. pp. 910-918 (2010)

[24] Rush, A.M., Chopra, S., Weston, J.: A neural attention model for abstractive sentence summarization. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015. pp. 379-389 (2015)

[25] See, A., Liu, P.J., Manning, C.D.: Get to the point: Summarization with pointergenerator networks. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers. pp. 1073-1083 (2017)

[26] Shen, D., Sun, J., Li, H., Yang, Q., Chen, Z.: Document summarization using conditional random fields. In: IJCAI 2007, Proceedings of the 20th International Joint Conference on Artificial Intelligence, Hyderabad, India, January 6-12, 2007. pp. 2862-2867 (2007)

[27] Wan, X., Yang, J.: Multi-document summarization using cluster-based link analysis. In: Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2008, Singapore, July 20-24, 2008. pp. 299-306 (2008)

[28] Yin, W., Pei, Y.: Optimizing sentence modeling and selection for document summarization. In: Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015. pp. 1383-1389 (2015)

[29] Yin, W., Pei, Y., Huang, L.: Automatic multi-document summarization based onnew sentence similarity measures. In: PRICAI 2012: Trends in Artificial Intelligence - 12th Pacific Rim International Conference on Artificial Intelligence, Kuching, Malaysia, September 3-7, 2012. Proceedings. pp. 832-837 (2012)

[30] Zeiler, M.D.: ADADELTA: an adaptive learning rate method. CoRR abs/1212.5701 (2012)