Short Text Representation Model Construction Method Based on Novel Semantic Aggregation Technology

Dong Yi[1(✉)], Zhai Jia[2], Li Xin[3], and Chen Feng[1]

[1] Science and Technology on Optical Radiation Laboratory, Beijing 100000, China
18810331577@sina.cn
[2] Science and Technology on Electromagnetic Scattering Laboratory, Beijing 100000, China
[3] Beijing Institute of Electronic System Engineering, Beijing 100000, China

Abstract. The semantic representation model of short texts has insufficient semantic representation ability, and the semantic representation method of short text based on the combination of word embedding and semantic weight is low in computational complexity and its performance is even better than that based on complex structure such as RNN and LSTM. This paper proposes a semantic representation model of short text based on ELMO (Embeddings from Language Models). The innovation of this model is: firstly, it is adopted the more advanced word embedding model ELMO; secondly, it is designed the semantic keyword extraction method of short text based on the topic model (Latent Dirichlet Allocation, LDA); thirdly, the stochastic gradient descent (SGD) is adopted, which is used to learn the semantic weights of semantic keywords in short texts. The experimental results show that compared with the existing short text semantic representation model, the representation model of short text, which is proposed in this paper, shows a high semantic representation ability of short text in specific domain and in the open domain.

Keywords: Short text representation model · ELMO · Topic model · Stochastic gradient descent

1 Introduction

Word vector can effectively capture the contextual semantic information and grammatical information of words, and it realizes the vectorized representation of words. It is a bridge for computer to understand human language. Therefore, various types of word embedding models emerge one after another, such as word embedding model based on statistical methods [1], word embedding model based on neural network language model word2vec [2] and it is recently proposed word

Supported by Science and Technology on Optical Radiation Laboratory.

H. Ning (Ed.): CyberDI 2019/CyberLife 2019, CCIS 1137, pp. 107–118, 2019.
https://doi.org/10.1007/978-981-15-1922-2_7

embedding model based on deep learning ELMO [3]. Vector representations of words are constantly emerging, which improves the semantic representations of word vectors. However, the current research on efficient vector representations for short texts (sentences, paragraphs, etc.) is still facing great challenges [4].

Currently, short text representation methods are based on complex networks (RNN, CNN) and word vector [11]. Le et al. [5] proposed an unsupervised text representation method (paragraph2vectors) that uses a method similar to Word2Vec [2], and can learn from variable length text fragments (such as sen-tences, paragraphs and documents) to a fixed length vector representation. Kiros et al. [6] proposed a generalized distributed sentence codec for unsupervised learning and trained the encoder-decoder model by using continuous text, which attempts to reconstruct sentences around the encoded paragraph and the sentences of share semantics and syntax information are mapped into a vector representation. Tai et al. [7] proposed a Tree-Lstm text semantic representation model based on tree structure, which introduced the standard LSTM structure into the tree structure network topology and achieved a superior sequence structure. The sentence vector representation capability of LSTM.

However, compared with the text representation method based on complex network, the methods based on word vector often have low computational complexity and satisfactory results. Generally, it can be achieved by averaging or maximizing the word vectors in short texts [8,9]. Wieting et al. [10] used a word vector and a semantic pair dataset to construct a text representation model by training the word average model. This method has excellent performance in natural language processing tasks, especially in text similarity, its performance is better than unweighted word vector averaging and even better than text representation models based RNN/CNN. Arora et al. [11] used a mainstream word vector representation model in unlabeled corpus (such as Wikipedia) to represent text by weighted averaging of word vectors, while using principal component analysis (PCA)/singular value decomposition (SVD) to fine-tuning, this text representation method improves the performance of text similarity measurement by about 10% to 30%. Boom et al. [12] constructed a short text representation model by weighted combination of inverse document frequency IDF and Wode2Vec and proved its validity in short text matching tasks. On this basis, a short text representation model based on Wode2Vec ($Word2Vec\_SGD$) was proposed, that is, each word in the short text is given a corresponding weight by a random gradient descent algorithm, and then the word vectors corresponding to the respective words in the short text are weighted and summed to obtain a vector representation of the short text.

Inspired by [12], this paper proposes a novel semantic aggregation technique based on the latest word vector generation model ELMO to construct a short text representation model. On the one hand, the semantic aggregation technique uses the LDA to extract the semantic keywords in the short text, thereby reducing the interference on words that are not related to the semantic expression of the short text, and reducing the computational redundancy in the subsequent training process of the semantic weight parameters; on the other hand, the Stochastic

Gradient Descent (SGD) is used to optimize the semantic keyword weights to give corresponding weights according to the importance of semantic keywords in short text semantic expression. The experimental results show that the proposed short text semantic representation model has excellent ability of semantic representation and domain adaptability.

2 Related Work

2.1 Word Embedding

ELMO [3] that is proposed recently can capture the semantic and syntactic information of words, and can also consider the situation in which words can express different meanings in the different context. Then, compared with the mainstream word vector model Word2Vec [2], it solves the problem of polysemy, and can obtain more accurate vector representation of words. The model is characterized by the fact that the characterization of each word is a function of the entire input. The specific method is to train the bidirectional long-term memory network model (bi-Lstm) with the language model as the target, and then use LSTM to generate the semantic vector of the words. The ELMO representation is "deep", that is, the word vector generated by ELMO is a function of the internal characterization of all layers of bi-Lstm to get the rich representation of words. The high-level LSTM can capture related features such as word semantics and context, while low-level LSTM can find grammatical features. Therefore, this paper will use the advanced word vector model ELMO to build a more semantic characterization ability of the representation model of short text.

2.2 LDA

LDA model is a bayesian unsupervised probability model with three-layer structure of word, topic and document, which can model the underlying topic information in the document [13]. The model makes the assumption that each word is extracted from a potential topic, each article is the probability distribution of the topic, and each topic is the probability distribution of the word.

Figure 1 shows the graph model of LDA, where V represents the number of dictionaries in the training corpus and M represents the number of documents in the training corpus, N_m represents the total number of words in m_{th} the document in the training corpus, and K represents the number of topics.θ_m represents the probability distribution of all topics in the m_{th} document,$Z_{m,n}$ represents the n_{th} topic in the m_{th} document, $W_{m,n}$ represents the n_{th} word of the m_{th} document, φ_K represents the probability distribution of all words in the n_{th} topic; θ_m is the Dirichulet prior distribution of super-parameter α, recorded as $\theta_m \sim \text{Dirichulet}(\alpha)$, φ_K is the Dirichulet prior distribution of super-parameter β recorded as $\varphi_K \sim \text{Dirichulet}(\beta)$.

The purpose of the LDA is to find potential topics in the document. It can be seen from Fig. 1 that the theme probability distribution θ_m (m = 1, 2, ...,

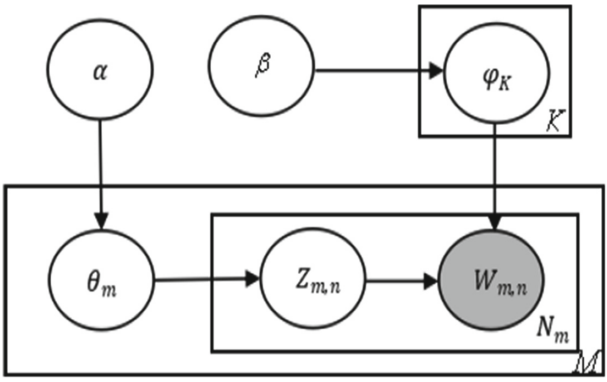

Fig. 1. The graph model of LDA

M) of the document is obtained according to the dirichulet prior distribution Dirichulet (α). Then, the probability distribution of each potential topic φ_K (k = 1, 2, ..., K) in the document is obtained according to the Dirichulet prior distribution. In other words, the generation process of each word $W_{m,n}$ (n = 1, 2, ...) in any document D_m (m = 1, 2, ...M). Extract a topic $Z_{m,n}$ from the multinomial distribution $Multi(\theta_m)$ corresponding to the document, and then extract a word $W_{m,n}$ from the multinomial $Multi(\varphi_K)$ corresponding to the topic $Z_{m,n}$. If the process is repeated N_m times, the document D_m is produced. This paper will use LDA's powerful text topic modeling ability to propose a short text semantic keyword extraction method based on LAD.

3 Short Text Representation Model Based on Novel Semantic Aggregation Technology

In order to improve the semantic representation ability of short text representation model, this paper adopts the advanced word vector model ELMO, and combines the semantic weighting scheme based on LDA and SGD to propose a novel short text representation model (STRM-SAT). The flow chart of the algorithm is shown in Fig. 2, including data preprocessing and semantic aggregation techniques.

3.1 Data Preprocessing

Data preprocessing is the first step of the STRM-SAT algorithm, which is mainly to perform lemmatization, word deduplication, and removing stop words on short texts. Then, for any short text $Text(w_1, w_2, \ldots, w_N)$, N represents the total number of words in the short text, and it is obtained the word sequence $Sequence_{word}(s_1, s_2, \ldots, s_M)$ about the short text after the data pre-processing. Where M is the number of words contained in the word sequence and $M \leq N$. This step mainly uses StanfordParser to do the above.

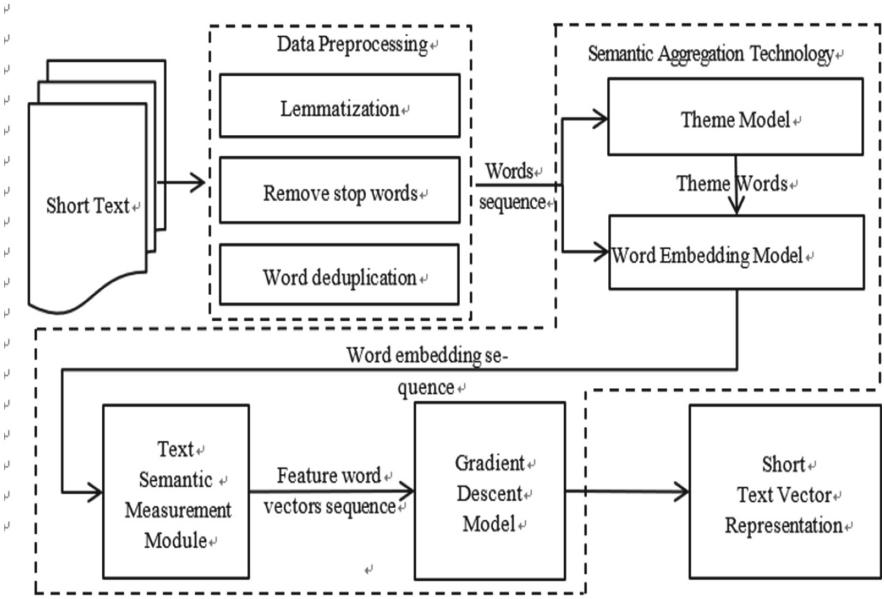

Fig. 2. Algorithm flow of STRM-SAT

3.2 Semantic Aggregation Technology

This paper proposes a novel semantic aggregation technique, which based on the advanced Word embedding model ELMO, and fused LDA and SGD, to construct a vector representation of short text. It should be pointed out that there are some words in the short text that are useless to their semantic expression or no clear semantic meaning. These words appear in many short texts, so there is more coincidence between non-related short texts. Deleting these words from short texts or reducing their impact helps to reduce their interference with the overall semantic expression of short texts. Based on this, the LDA is introduced in the new semantic aggregation technology to design a short text semantic keyword extraction method based on LDA. On the other hand, the SGD is introduced to design a keyword semantic weight learning mechanism based on SGD.

Short Text Semantic Keyword Selection Mechanism Based on LDA. The LDA can learn potential topic information from largescale corpus. Then, the topic words is obtained through LDA for any short text, those topic information can be regarded as a highsummary expression of short text semantic information. Therefore, the semantic distance must be close between a word, which plays a key role in the semantic expression of a short text, and the sequence of the topic words.

Based on above, this paper constructs a semantic keyword extraction method based on LDA to obtain semantic keywords in short text. The specific calculation

steps are as follows: firstly, the topic word sequence $Sequence_{toptic}(t_1, t_2, \ldots, t_K)$ of the corresponding short text is obtained through the trained LDA model, where K represents the number of topic words, and then the word vector sequence F and H about A and B are respectively obtained according to the trained ELMO; then, it is calculated the semantic distance between $s_m(0 < m \leq M)$ and $Sequence_{toptic}$, ie.

$$Dis = \frac{1}{K} \sum_{k=1}^{K} \frac{v_{sm} \cdot v_k}{|v_{sm}| \times |v_k|} \tag{1}$$

Therefore, the semantic distance between each word in the short text and $Sequence_{toptic}$ is sequentially calculated by formula (1) to determine the semantic keyword sequence $Sequence_{features}(f_1,f_2,...,f_H)$ of the short text, where H represents the total number of semantic keywords. After many experiments, it is verified that H takes 20, and the words in $Sequence_{features}$ are arranged in descending order according to the semantic distance.

Keyword Semantic Weight Learning Mechanism Based on SGD. Through the above steps, the semantic keywords of short text can be obtained. However, the semantic keywords in $Sequence_{features}$ are different in the semantic expression of short text. Therefore, this paper uses the machine learning algorithm to learn the corresponding weighting factors β_g of semantic keywords, g $\subseteq$ [1, H], in the semantic expression of short text from the large-scale corpus to obtain the short text semantic vector. The specific idea is as follows: As shown in Fig. 3, the vector representation sequence $Vec\ (v_{f_1}, v_{f_2}, ..., v_{f_H})$ of $Sequence_{features}(f_1, f_2, ..., f_H)$ is obtained by the trained ELMO model. Next, v_{f_g} is multiplied by its corresponding weighting factor β_g, and summing and averaging to obtain the feature vector of the short text. The calculation formula is as shown in (2):

$$V = \frac{1}{H} \sum_{g=1}^{H} \beta_g \cdot v_{f_g} \tag{2}$$

In order to learn the weighting factor β_g in Eq. (2), a loss function is defined in this paper. For any short text pairs $p(V_1, V_1)$, if p is semantically related, maximize the semantic similarity between short texts in p; if p is semantically uncorrelated, minimize the semantic similarity between short texts in p:

$$f(p) = \begin{cases} SC(V_1, V_2) & , if\ p\ is\ related \\ -SC(V_1, V_2) & , if\ p\ is\ unrelated \end{cases} \tag{3}$$

SC(·) is a function to measure the semantic distance between two short texts. This paper uses the cosine of the short text feature vector to measure the semantic distance:

$$SC(V_1, V_2) = \frac{V_1 \cdot V_2}{V_1 \times V_2} \tag{4}$$

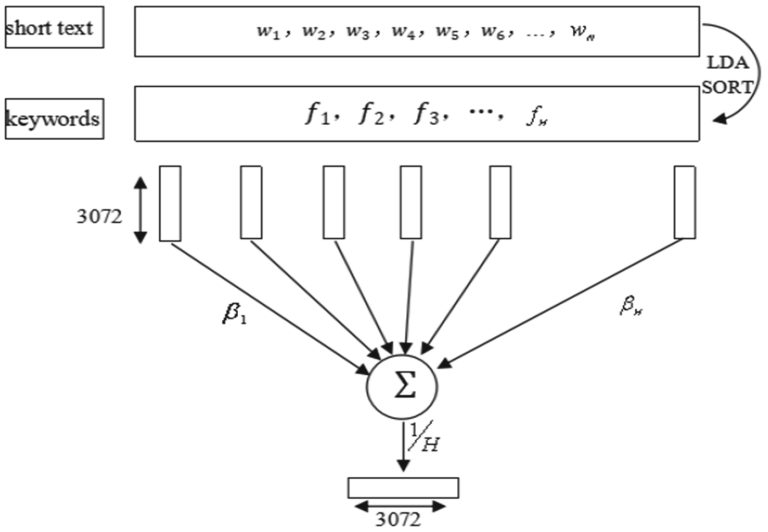

Fig. 3. The calculation process of short text semantic aggregation

Next, the paper constructs the following objective function of the weighting factor:

$$S(\beta_1, \beta_2, ..., \beta_h) = \frac{1}{|D|} \sum_{p \subseteq D} f(p) + \lambda \sum_{j=1}^{h} \beta_j^2 \tag{5}$$

where the corpus D is composed of short text pairs and the number of semantically related short text pairs is the same as the number of non-semantic related short text pairs, and $|D|$ represents the total number of short text pairs in D. In order to maximize the objective function, this paper uses the stochastic gradient descent algorithm (SGD). Figure 4 shows the changes in the semantic weighting factors, which obtained by SGD. Obviously, as the index of semantic keywords increases, the value of the weighting factor decreases gradually. This indicates that the closer the semantic distance between the sequence of theme words of short text and keyword, the weighting factor of this keyword is the larger, so it is more important in the semantics expression of the short text.

4 Experiment and Result Analysis

Next, the short text matching task is used to verify the validity of the short text representation model STRM-SAT proposed in this paper. The performance of STRM-SAT in specific fields and open fields will be verified on the self-built corpus and the public corpus respectively.

The control methods we used are described as follows:

XXX_ Mean: the model of short text representation is constructed by adding and averaging the word vectors in the short text.

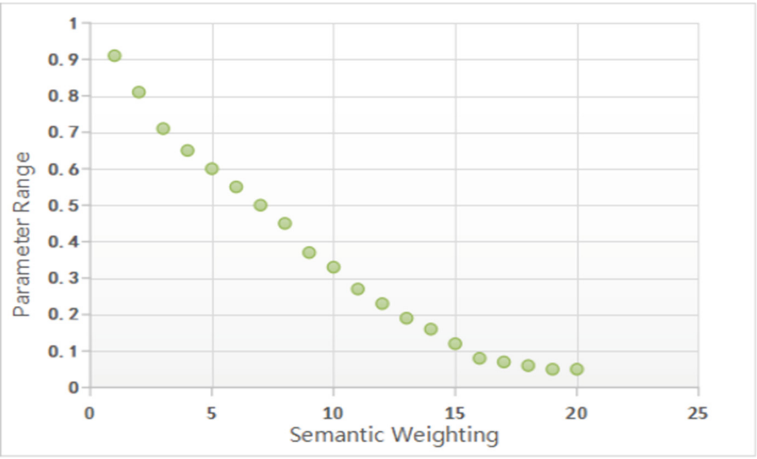

Fig. 4. Trend of semantic weighting factor

XXX_Idf: Constructing a short text representation model by using the inverse document frequency (idf) of each word in the short text as the weight and the word vector is multiplied by the corresponding weight to be added and averaged.

XXX_Top30%_Idf: the words in the short text are sorted according to their idf values from large to small, and the word vectors corresponding to the first 30% of the words are multiplied by the corresponding idf to be added and averaged, and the short text representation model is constructed.

Among them, "XXX" represents the word vector model, we use Word2Vec, ELMO_3072 and ELMO_1024 respectively. At the same time, we also use Word2Vec_SGD for control experiment.

4.1 Short Text Matching Experiment Based on Domain Corpus

The corpus used in this experiment was crawled from the official website of PubMed and the official website of Journal of Neuroscience, mainly to verify the semantic representation ability of the short text representation models in the biomedical literature corpus.

Original corpus: The corpus used in this paper consists of two parts, one is the abstract data set $Corpus_{pubMed}$ from the PubMed official website, and the other is the full-text data set $Corpus_{neurosc}$ from the Journal of Neuroscience.

LDA training corpus consists of abstracts from the fields of depression, epilepsy, cytology, clinical medicine, and computer science in $Corpus_{pubMed}$.

Building a short text pairs corpus: the summaries of the papers associated with the depression or depression drug entity is extracted from $Corpus_{pubMed}$ to form a set A, and then the set B is consisted of a summary of different topics in A is then extracted from the $Corpus_{pubMed}$. Next, calculate the correlation of short text pairs based on the method described in [10]. The rules of the construction

of Corpus: firstly, calculate the correlation of any two short text pairs in A. If the value is greater than 0.7, mark it as a semantically related short text pair and join the $Corpus_{pairs}$. Then, take a short text from each of A and B and calculate the correlation. If the value is less than 0.3, mark it as a non-semantic related pair and add it to the $Corpus_{pairs}$. Finally, semantically related pairs and non-semantic related pairs take 50,000 each to form the final $Corpus_{pairs}$. $Corpus_{pairs}$ is divided into training set TS_1 and test set TS_2 according to 4:1, where TS_1 is used to train SGD and TS_2 is used for final short text matching experiment.

The training corpus of ELMO and Word2Vec is composed of $Corpus_{pubMed}$ and $Corpus_{neurosc}$. In addition, the dimension of ELMO adopts the three-layer feature and top-level feature respectively, and the corresponding vector dimension is 3072 and 1024, respectively, which are recorded as ELMO_3072 and ELMO_1024, respectively. The dimension of the Word2Vec word vector is 300. Then, the results of this experiment are shown in Table 1.

Table 1. Comparison of experimental results.

No	Algorithm	Precision
1	Word2Vec_Mean	73.87%
2	Word2Vec_Idf	71.13%
3	Word2Vec_Top30%_Idf	78.39%
4	Word2Vec_SGD	80.59%
5	ELMO_1024_Mean	74.12%
6	ELMO_1024_Idf	70.87%
7	ELMO_1024_Top30%_Idf	80.31%
8	**STRM-SAT_1024**	81.17%
9	ELMO_3072_Mean	73.57%
10	ELMO_3072_Idf	74.67%
11	ELMO_3072_Top30%_Idf	80.74%
12	**STRM-SAT_3072**	83.32%

According to Table 1, the model of short text representation using ELMO is better performance than the model of short text representation using Word2Vec in the task, and the higher the dimension of the ELMO, the better the performance of the short text representation model, which shows that on the one hand, the word vector generated by ELMO has more semantic representation ability than the word vector generated by Word2Vec; on the other hand, the higher the dimension of ELMO, the richer the semantic information can be captured and the more powerful the semantic representation ability.

In this experiment, XXX_Top30%_Idf improved 5%–7% performance compared to XXX_Mean, while Word2Vec_SGD and STRM-SAT with finer semantic

weighting schemes showed higher performance, on the one hand, the weighted combination of word vector and inverse document frequency is effective, on the other hand, the weighting scheme used in Word2Vec_SGD and STRM-SAT uses the machine learning method to obtain more accurate weights, therefore, so, the better performance has been achieved.

Compared with Word2Vec_SGD, STRM-SAT performed better in this experiment. On the one hand, STRM-SAT eliminates these words that are useless or no clear semantic meaning for short text semantic expressions through the LAD. These words appear in many short texts, so there is more coincidence between unrelated short texts. These words are deleted from short text or reduced their impact to helps to increase the values of similarity between similar short text pairs and reduces the values of similarity between non-similarity short texts. On the other hand, STRM-SAT adopts a more advanced word vector model ELMO, which not only can effectively capture the semantics of words and the grammar of words, but also can generate corresponding word vector representations according to the meaning of words in different contexts. Therefore, the word vectors, which is generated ELMO, are of higher quality, which is critical to the semantic representation of the STRM-SAT.

4.2 Short Text Matching Experiment Based on Open Domain Corpus

ELMO: The model is from the official website of ELMO (https://allennlp.org/elmo). ELMO's training corpus is from Wikipedia (1.9B) and WMT 2008–2012 (3.6B). The dimensions of the ELMO used in this paper are 3072 and 1024, respectively, which are recorded as ELMO_3072 and ELMO_1024, respectively.

Word2Vec: The model comes from its official website (http://code.google.com/archive/p/word2vec/), its training data comes from the Google News dataset (100 million words), and the dimension of vector is 300.

The training data of LDA uses the Wikipedia corpus used in [14]. The SGD training corpus uses the SemEval Semantic Text Similarity Task (2012–2015) data set used in [11]. The test data used in this experiment were from the SemEval Twitter task [15] and the SemEval semantic relevance task [16]. The experimental results are shown in Table 2.

As can be seen from Table 1, Word2Vec_SGD, STRM-SAT_1024, and STRM_SAT_3072 achieved good results, which is consistent with the results of short text matching experiments based on specific domain corpus. Further analysis shows that the weighted word vector method exhibits better semantic representation ability than the unweighted word vector method, and the vector representation of short text is obtained by way of the machine learning based semantic weighting scheme that is the best semantic representation ability. In addition, the STRM-SAT proposed in this paper has achieved the best results in the experiment due to the more effective word vector model ELMO and the semantic keyword extraction method based on machine learning.

Through the above experiments, the performance of STRM-SAT proposed in this paper is higher than other comparison methods, whether it is in specific

Table 2. Comparison of experimental results.

No	Algorithm	Precision
1	Word2Vec_Mean	80.10%
2	Word2Vec_Idf	80.42%
3	Word2Vec_Top30%_Idf	81.11%
4	Word2Vec_SGD	84.42%
5	ELMO_1024_Mean	78.23%
6	ELMO_1024_Idf	81.67%
7	ELMO_1024_Top30%_Idf	81.71%
8	**STRM-SAT_1024**	86.37%
9	ELMO_3072_Mean	79.88%
10	ELMO_3072_Idf	81.17%
11	ELMO_3072_Top30%_Idf	84.55%
12	**STRM-SAT_3072**	87.11%

domain or in the open domain test corpus. This shows the superiority of STRM-SAT, and also shows that STRM-SAT has strong domain adaptability.

5 Conclusion

This paper explores the semantic aggregation technology based on the advanced word vector generation model ELMO to construct the short text semantic representation model STRM-SAT, and designs the short text semantic keyword extraction method based on LDA and the keyword semantic weight learning mechanism based on SGD, which tries to combine the semantic information of the word vector in an optimal way to realize the Precise expression of short text semantic information. The order information of word plays an important role in semantic expression of short text. Therefore, in the future work, we will try to integrate the order information of word into the vector representation model of short text to realize the all-round modeling of short text from semantic to word order.

References

1. Bengio, Y., Ducharme, R., Vincent, P., Janvin, C.: A neural probabilistic language model. J. Mach. Learn. Res. **3**, 1137–1155 (2003). ISSN 15324435. https://doi.org/10.1162/153244303322533223
2. 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, pp. 3111–3119 (2013)
3. Peters, M.E., et al.: Deep contextualized word representations. In: Proceedings of NAACL (2018)

4. Conneau, A., Kiela, D., Schwenk, H., Barrault, L., Bordes, A.: Supervised learning of universal sentence representations from natural language inference data. In: EMNLP, pp. 670–680. Association for Computational Linguistics (2017)
5. Le, Q.V., Mikolov, T.: Distributed Representations of Sentences and Documents. arXiv.org, May 2014
6. Kiros, R., Zhu, Y., Salakhutdinov, R.R., Zemel, R., Urtasun, R., Torralba, A., Fidler, S.: Skip-thought vectors. In: Advances in Neural Information Processing Systems (2015)
7. Tai, K.S., Socher, R., Manning, C.D.: Improved semantic representations from tree-structured long short-term memory networks. arXiv preprint arXiv:1503.00075 (2015)
8. Weston, J., Chopra, S., Adams, K.: #TagSpace: semanticembeddings from hashtags. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP) (2014)
9. dos Santos, C.N., Gatti, M.: Deep convolutional neural networks for sentiment analysis of short texts. In: The 25th International Conference on Computational Linguistics, COLING 2014, Dublin, pp. 69–78, July 2014
10. Wieting, J., Bansal, M., Gimpel, K., Livescu, K.: Towards universal paraphrastic sentence embeddings. In: International Conference on Learning Representations (2016)
11. Arora, S., Liang, Y., Ma, T.: A simple but tough-to-beat baseline for sentence embeddings (2017)
12. Boom, C.D., Canneyt, S.V., Bohez, S., et al.: Learning semantic similarity for very short texts, pp. 1229–1234 (2015)
13. Ghassabeh, Y.A., Rudzicz, F., Moghaddam, H.A.: Fast incremental LDA feature extraction. Pattern Recogn. **48**(6), 1999–2012 (2015)
14. De Boom, C., Canneyt, S.V., Demeester, T., et al.: Representation learning for very short texts using weighted word embedding aggregation. Pattern Recogn. Lett. **80**, 150–156 (2016)
15. Xu, W., Callison-Burch, C., Dolan, W.B.: SemEval-2015 task 1: paraphrase and semantic similarity in Twitter (pit). In: Proceedings of SemEval (2015)
16. Marelli, M., Bentivogli, L., Baroni, M., Bernardi, R., Menini, S., Zamparelli, R.: SemEval-2014 task 1: evaluation of compositional distributional semantic models on full sentences through semantic relatedness and textual entailment. In: SemEval-2014 (2014)