

An Improved Method for Web Text Affective Cognition Computing Based on Knowledge Graph

Bohan Niu^{1,*} and Yongfeng Huang²

Abstract: The goal of research on the topics such as sentiment analysis and cognition is to analyze the opinions, emotions, evaluations and attitudes that people hold about the entities and their attributes from the text. The word level affective cognition becomes an important topic in sentiment analysis. Extracting the (attribute, opinion word) binary relationship by word segmentation and dependency parsing, and labeling those by existing emotional dictionary combined with webpage information and manual annotation, this paper constitutes a binary relationship knowledge base. By using knowledge embedding method, embedding each element in (attribute, opinion, opinion word) as a word vector into the Knowledge Graph by TransG, and defining an algorithm to distinguish the opinion between the attribute word vector and the opinion word vector. Compared with traditional method, this engine has the advantages of high processing speed and low occupancy, which makes up the time-costing and high calculating complexity in the former methods.

Keywords: Affective cognition, fine-grained, knowledge representation, knowledge graph.

1 Introduction

Affective cognition, also known as sentiment analysis or opinion mining, aims to analyze the content of people's emotions, opinions, evaluations and attitudes expressed by entities and their attributes. The entities involved are very extensive and can be products, services, institutions, individuals, events, problems, topics, and so on. Because viewpoint information is very important to people's actions and behaviors: whether they are individuals or collectives, they often seek opinions and suggestions from others when making decisions. Therefore, the analysis of viewpoint information has a very wide practical significance.

The number of evaluations for a product is often very large, and the number of words is too long. It is almost impossible for an individual or a business to completely read it carefully. Selecting a comment for analysis tends to ignore some details or have a personal tendency to make users or businesses unable to do so. Get objective and comprehensive feedback. Therefore, an intuitive and efficient network text sentiment

¹ Department of Electrical Engineering, Columbia University, New York, NY 10027, USA.

² Department of Electronic Engineering, Tsinghua University, Beijing, 100084, China.

* Corresponding Author: Bohan Niu. Email: bn2300@Columbia.edu.

analysis mechanism is needed to analyze the reviews. By analyzing the emotions expressed by the product attributes and emotional words in the review text, the user can intuitively and quickly understand the advantages and disadvantages of various attributes of the goods without having to read all the reviews, thereby allowing the user to have a more comprehensive view of the entire product. At the same time, for merchants, it is possible to more quickly re-design or improve the parts that are not highly evaluated according to the user's feedback on the product attributes, so that the merchants can better grasp the market.

The concept of sentiment analysis was first proposed by Hatzivassiloglou et al. in 1997 [Hatzivassiloglou and McKeown (1997)]. After that, the related technologies and applications of sentiment analysis have developed rapidly. With the rise and popularity of social media in recent years, a number of domestic and international top conferences have included the sentiment analysis of web texts as the theme. In 2008, Blair-Goldensohn et al. [Blair-Goldensohn, Hannan, McDonald et al. (2008)] proposed a general model of attribute-opinion relationship extraction for service-oriented reviews. The model uses sentence/phrase level emotion classification, attribute-opinion extraction and clustering processes. Kim et al. [KIM and Hovy (2004)] proposed a 4-tuple model: [Topic, Holder, Claim, Sentiment], i.e., (subject, opinion holder, statement, opinion). Liu et al. [Liu and Zhang (2012)] proposed a 5-tuple model (entity/subject, feature/aspect/attribute, sentiment polarity, publisher, publication time). Jin et al. [Jin, Ho and Srihari (2009)] proposed a novel machine learning method based on a lexicalized HMM framework that integrates multiple important language features which can predict new potential product and opinion entities based on the learned patterns. Su et al. [Su, Xu, Guo et al. (2008)] proposed a mutually reinforcing method to solve the problem of extracting opinions, able to cluster and optimize product features and opinion words, and construct a set of the words and product features, and combine the polarity dictionaries to distinguish the opinion of the set. Brody et al. [Brody and Elhadad (2010)] proposed a simple and flexible unsupervised extraction algorithm, which extracts product features by setting a certain topic and discriminates emotional tendencies based on positive and negative emotional word seed sets. With the rise of deep learning and neural networks, many researchers in the field of sentiment analysis have also applied it to the sentiment analysis of online texts. In 2015, Liu et al. [Liu and Chen (2015)] obtained opinions and attitudes on hot topics among microblog users through Convolutional Neural Network (CNN). By using a CNN, the problem of explicit feature extraction and implicit learning in training data is solved. Socher et al. [Socher, Perelygin, Wu et al. (2013)] proposed the re-cursive tensor neural network (RNTN) to solve the problem that the long sentences in the previous model could not be effectively interpreted by the semantic space. The accuracy of RNTN in sentiment prediction reached 80.7%, surpassing the previous model. Yang et al. [Yang, Tu, Wang et al. (2017)] proposed an attention-based long-term memory model (Attention-based LSTM) to improve the accuracy of object-dependent sentiment classification. The accuracy of the algorithm is improved by learning the distance between the target entity and its most significant feature.

It can be seen from the current research situation that previous scholars have conducted a lot of research on sentiment analysis in such fields as computational linguistics, cognitive psychology, natural language processing, and data mining. In this paper, we will use a

different approach from previous studies, which applies the knowledge representation learning and TransG model to the field of affective cognition, and to establish a comprehensive template rule for the extraction of binary relationships, with the structure of (attribute, opinion words) and will act as the smallest emotional expression unit. The method of knowledge representation learning transforms attributes and opinion words into word vectors. Then, the word vectors calculated by the model constitute a binary relationship knowledge map, and the derived knowledge map is accessed to web pages. This method solves the problems of long time and high complexity in the previous sentiment analysis methods which make it possible to complete the online sentiment analysis and processing tasks based on the web.

2 Related works

At present, most of the word-level affective cognition systems are constituted mainly by the emotional dictionary, which divides the words into positive emotional words and negative emotional words and is stored in the positive sentiment dictionary and the negative sentiment dictionary respectively. We can regard this representation as expressing a sentiment word and its collocation as a vector, except that the vector has only one dimension that is non-zero, while others are all zero, which we usually call this representation one-hot representation. This representation is very simple, no learning process, and widely used in information retrieval and natural language processing. However, the one-hot representation method has obvious drawbacks, that is, all objects are assumed to be independent of each other, which means the vectors of all objects are orthogonal to each other [Turian, Ratinov and Bengio (2010)]. After learning the main methods of knowledge representation, we found that the Translation model has a high degree of fit for this study as well as has a simple model. Therefore, the subsequent research mainly focuses on the Translation model.

In 2013, Mikolov et al. [Mikolov, Chen, Corrado et al. (2013)] used word2vec to represent the learning model and found that there is a very common translation invariance in the word vector space:

$$\mathbf{C}(\textit{king}) - \mathbf{C}(\textit{queen}) \approx \mathbf{C}(\textit{man}) - \mathbf{C}(\textit{woman}) \quad (1)$$

Where $\mathbf{C}(W)$ represents the word vector of the W word obtained using the word2vec model. This phenomenon indicates that the word vector can find that there is some similar implicit semantic relationship between the word *king* and *queen*, *man* and *woman*, and the implicit and similar semantic relationship between the words exists widely in the vocabulary. Inspired by this phenomenon, the researchers have successively proposed the Translation model. Bordes et al. [Bordes, Usunier, Garcia-Duran et al. (2013)] proposed the TransE model in 2013, and the vector \mathbf{l}_r of the relationship r in each triple (h, r, t) is translated as the head entity vector \mathbf{l}_h and the tail entity vector \mathbf{l}_t , that is, for each the triads (h, r, t) compared with the previous knowledge representation learning model, the TransE model has great advantages in terms of computational complexity and number of parameters. Especially, on large-scale sparse knowledge maps, the performance of TransE is even more amazing. In the test with Wordnet as the data set, TransE's accuracy rate (HITS@10) reached 75.4%, far exceeding the results of the previous knowledge representation learning model.

Although TransE has many advantages, it also has obvious shortcomings. Because the TransE model is too simple, it often fails to deal with the complex relationship (one-to-many, many-to-one, many-to-many) of the knowledge base. Because in complex relationships, we can easily understand: If the relationship r is a many-to-one relationship, we can easily get $\mathbf{l}_{h_0} \approx \mathbf{l}_{h_1} \approx \dots \approx \mathbf{l}_{h_m}$, in the face of the relationship r is a pair the opposite of multiple problems. This deficiency will have a very large impact on the correct rate of the model.

Wang et al. [Wang, Zhang, Feng et al. (2014)] proposed the TransH model in 2014 to address the shortcomings of the TransE model mentioned earlier. The main idea is to project the head and tail entity vectors in different relationships into different spaces. In addition, the number of head and tail entities corresponding to the same relationship is not necessarily the same in complex relationship problems. Therefore, when the corrupted triples are generated, the probability of the head and tail entities is not randomly replaced as in TransE, but by the probability determined by the number of the head and tail entities. The test results in Wang et al. [Wang, Zhang, Feng et al. (2014)] show that in Freebase15k dataset, where the relationship is relatively complex, TransH's accuracy for link predictions (HITS@10) is as high as 64.4%, which is much higher than that of TransE's 58.5%, reflecting the superiority of the TransH model for complex relationships.

Lin et al. [Lin, Liu, Sun et al. (2015)] believe that the TransH model's hypothesis of placing entities and relations in the same semantic space limits the accuracy of TransH to some extent. In order to overcome TransH's shortcomings, they propose TransR model, in which different relationships have different semantic spaces and project entities of different relationships to different semantic spaces.

Ji et al. [Ji, He, Xu et al. (2015)] believe that although the TransR model makes up some shortcomings of the TransE and TransH models, there are still limits: in the same relationship, the head and tail entities share the same projection matrix. However, the attributes or types of the head and tail entities of a relationship are often different, even huge. The projection from the entities semantic space to the relation semantic space is the result of the interaction of the entities and the relations, so it is unreasonable that the projection matrix is only related to the relationship in the previous model.

Due to the introduction of spatial projection, TransR has a sharp increase in model parameters compared to TransE and TransH, which greatly increases the computational complexity of the algorithm. In order to solve these problems, Ji et al. [Ji, He, Xu et al. (2015)] proposed the TransD model. For a given triple (h, r, t) , the TransD model sets two projection matrices \mathbf{M}_{rh} and \mathbf{M}_{rt} respectively to project the head and tail entities to the relation space. The use of the projection matrix set by the two projection vectors solves the problem of too many parameters in the TransR model.

Xiao et al. [Xiao, Huang, Hao et al. (2015)] think that the loss function including TransE and the improved model is too simple, considering each dimension of the entity and the relation vector in the same dimension, which reduces the accuracy to some extent.

The TransA model changes the distance metric in the loss function from the or distance to the Mahalanobis distance and sets a weight matrix \mathbf{W}_r . After using Freebase15k to check the accuracy of the model, the TransA model's triplet prediction accuracy reaches

80.4%, and the accuracy of the Wordnet18 dataset is 94.3%, which is much higher than all previous models.

The TransG model, proposed by Xiao et al. [Xiao, Huang and Zhu (2016)], first takes the multi-semantic problem in the relationship in to consideration. The accuracy of the TransG model on the Freebase15k dataset is 88.2%, and the accuracy on the Wordnet18 dataset is 94.9%, which is a significant improvement over the previous model.

3 An improve method for Web text affective cognition computing

In order to simplify the model, this paper introduces the Translation method in the knowledge graph into the word-level sentiment analysis, which greatly simplifies the model and various parameters required for training by classifying the relationship between the words while vectorising the words. Taking into account the different semantic characteristics of the same emotion, the paper finally chooses to use the TransG model to replace the input (entity, relationship, entity) triples with (attribute, opinion, opinion word) triples and generate The corrupt triples by self-sampling method. After the TransG model, we can get the vector of each word and those words compose the knowledge graph. By analyzing the relationship between the word vector of attribute and opinion word, we could obtain the opinion they represent. The process of generating the knowledge map is shown in Fig. 1.

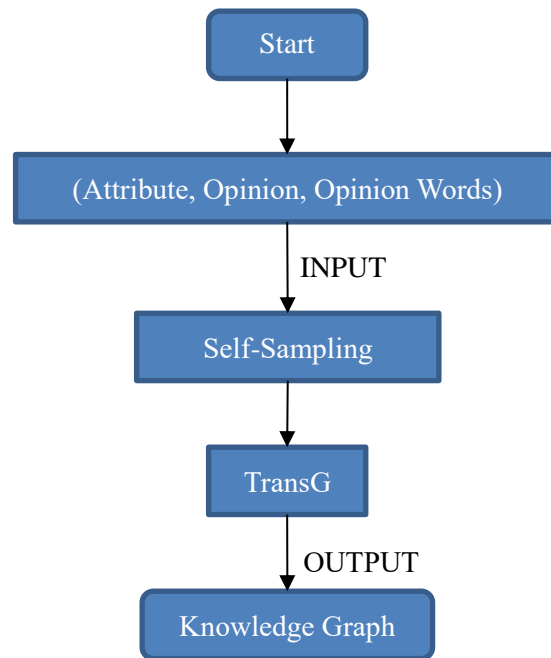


Figure 1: The generation of knowledge graph

3.1 Binary relationship extraction

The Language Technology Platform (LTP) is used as a word segmentation tool, which provides a Python library, called pyltp, that can be easily integrated into existing projects

and provides fast and high-accuracy Chinese word segmentation.

In addition to the word segmentation function, the language technology platform also provides a dependency syntax analysis module, which reveals its syntactic structure by analyzing the dependencies between components in a language unit. The main idea is to use the core verb of a sentence as the origin, and the other sentences. The components are dependent on the core verbs in some grammatical relations. At the same time, they regard the sentences as a dependent syntax tree. The nodes of the tree represent different words, and the edges of the trees represent dependencies, thus reflecting the dependence between words.

3.2 Improvement of self-sampling

From TransE, in order to optimize the accuracy for complex problems, self-sampling is used to generate corrupted triples for training:

$$S'_{(h,r,t)} = \{(h', r, t) | h' \in E\} \cup \{(h, r, t') | t' \in E\} \quad (2)$$

While in this project, the effect of the above method is greatly limited. Since the head and tail of the original model are selected from the entity library, which includes all the head and tail entities, the difference between the head and tail entities is so large that such random replacement is easy to generate a corrupted triple that does not contain any useful information. For example, in a set of entities with positive relationship in this article, it contains: (cost-effective, high, price, low). For the golden triple (cost-effective, positive, high), it is easy to generate a corrupted triple (high, positive, low) by Eq. (2), which contains no useful information, reducing the utilization of data. Due to this reason, we proposed an improved self-sampling method:

$$S'_{(h,l,t)} = \{(h', l, t) | h' \in H\} \cup \{(h, l, t') | t' \in T\} \quad (3)$$

where H is the set of head entities, that is, all the attributes; T is the set of tail entities, all the opinion words, $H \in E, T \in E$.

By Eq. (3), the above golden triple (cost-effective, positive, high) can only generate corrupted triple (cost-effective, positive, low) or (price, positive, high). The probability determined by Eq. (4), where hpt means the average number of head entities per tail entities, among all the relations; tph converts [Wang, Zhang, Feng et al. (2014)]. In this way, the use of data has significantly increased.

$$P(\text{changing head}) = \frac{tph}{tph+hpt} \quad (4)$$

3.3 Generating knowledge graph

With the golden triples and corrupted triples, this paper uses the TransG to generate knowledge graph. The TransG uses a Bayesian non-parametric infinite mixture embedding model, which is generally as follows:

(1) For an entity $e \in E$, initialize the entity vector which mean vector follows a standard normal distribution: $\mathbf{u}_e \sim \mathcal{N}(0,1)$.

(2) For a triple (h, r, t) :

(a) Extracting semantic components by Chinese Restaurant Processes: $\pi_{r,m} \sim CRP(\beta)$.

(b) Initialize the head vectors which mean follows a standard normal distribution: $\mathbf{h} \sim \mathcal{N}(\mathbf{u}_h, \sigma_h^2 \mathbf{E})$.

(c) Initialize the tail vectors which mean follows a standard normal distribution: $\mathbf{t} \sim \mathcal{N}(\mathbf{u}_t, \sigma_t^2 \mathbf{E})$.

(d) Get the opinion vectors: $\mathbf{u}_{r,m} = \mathbf{t} - \mathbf{h} \sim \mathcal{N}(\mathbf{u}_t - \mathbf{u}_h, (\sigma_h^2 + \sigma_t^2) \mathbf{E})$.

where \mathbf{u}_h and \mathbf{u}_t are the mean embedded word vector of attributes and opinion words, respectively, σ_h^2 and σ_t^2 are the variances of corresponding the attribute and the opinion word, and $\mathbf{u}_{r,m}$ is the m -th semantic word vectors. By using the Chinese Restaurant Process (CRP), TransG can automatically detect the different semantics of the same relationship, the different uses of the same opinion in this paper. In this setting, we can define the score function:

$$\mathbb{P}\{(h, r, t)\} = \sum_{m=1}^{M_r} \pi_{r,m} e^{-\frac{\|\mathbf{u}_h + \mathbf{u}_{r,m} - \mathbf{u}_t\|_2^2}{\sigma_h^2 + \sigma_t^2}} \quad (5)$$

where $\pi_{r,m}$ is a weight factor represents the weight of the i -th component, and M_r is the total number of semantics of the opinion r learned from the CRP.

In the previous model, when the word vector of the relationship r was determined, the geometric representation of the triple (h, r, t) was also fixed in the form $\mathbf{l}_h + \mathbf{l}_r \approx \mathbf{l}_t$. While in TransG, the geometric representation of the triple (h, r, t) is changed to:

$$m_{(h,r,t)}^* = \arg \max_{m=1 \dots M_r} \left(\pi_{r,m} e^{-\frac{\|\mathbf{u}_h + \mathbf{u}_{r,m} - \mathbf{u}_t\|_2^2}{\sigma_h^2 + \sigma_t^2}} \right) \quad (6)$$

$$\mathbf{h} + \mathbf{u}_{r,m_{(h,r,t)}^*} = \mathbf{t} \quad (7)$$

where $m_{(h,r,t)}^*$ indicates the m -th semantic of the current relation r . When given a triple (h, r, t) , the TransG model first finds semantics the relationship r belongs to, and then translate the header vector and the tail vector into the knowledge graph.

For most triples, only one semantic is at $\left(\pi_{r,m} e^{-\frac{\|\mathbf{u}_h + \mathbf{u}_{r,m} - \mathbf{u}_t\|_2^2}{\sigma_h^2 + \sigma_t^2}} \right)$ contains a non-zero value with a huge absolute value. Other semantics will become very small due to exponential decay. That is, when $m \neq m_{(h,r,t)}^*$, $\left(\frac{\|\mathbf{u}_h + \mathbf{u}_{r,m} - \mathbf{u}_t\|_2^2}{\sigma_h^2 + \sigma_t^2} \right)$ will become so large that the exponential function will get a very small value. In this way, noise generated from other semantics can be effectively ignored, and automatically select the semantics component of the relationship that best fits the triplet.

During the training process, the maximum data likelihood principle was used. For the non-parametric part, the weight matrix (h, r, t) is generated by the CRP. For a triple (h, r, t) , the probability of generating a new semantic component is defined as follows:

$$\mathbb{P}(m_{r,new}) = \frac{\beta e^{-\frac{\|h-t\|_2^2}{\sigma_h^2 + \sigma_t^2 + 2}}}{\beta e^{-\frac{\|h-t\|_2^2}{\sigma_h^2 + \sigma_t^2 + 2}} + \mathbb{P}\{(h,r,t)\}} \quad (8)$$

where $\mathbb{P}\{(h,r,t)\}$ is the currently calculated posterior probability. In order to better distinguish the correct triples from the wrong triples, this model maximizes the likelihood ratio of the golden triples to the corrupted triples. Combining all the conditions mentioned above, the training objective function of the model is as follows:

$$\begin{aligned} \min_{\mathbf{u}_h, \mathbf{u}_r, \mathbf{u}_t} \mathcal{L}, \mathcal{L} = & - \sum_{(h,r,t) \in \Delta} \ln \left(\sum_{m=1}^{M_r} \pi_{r,m} e^{-\frac{\|\mathbf{u}_h + \mathbf{u}_r, m - \mathbf{u}_t\|_2^2}{\sigma_h^2 + \sigma_t^2}} \right) + \\ & \sum_{(h',r,t') \in \Delta'} \ln \left(\sum_{m=1}^{M_r} \pi_{r,m} e^{-\frac{\|\mathbf{u}_{h'} + \mathbf{u}_r, m - \mathbf{u}_{t'}\|_2^2}{\sigma_{h'}^2 + \sigma_{t'}^2}} \right) \\ & + C \left(\sum_{r \in R} \sum_{m=1}^{M_r} \|\mathbf{u}_{r,m}\|_2^2 + \sum_{e \in E} \|\mathbf{u}_e\|_2^2 \right) \end{aligned} \quad (9)$$

where Δ is a collection of golden triples, Δ' is a collection of corrupted triples, C controls the degree of scaling, E is the set of entities, R is the set of relations, and the weights $\pi_{r,m}$ and variance σ are also learned from the optimization of the objective function.

In this model, we use stochastic gradient descent (SGD) to solve the optimization problem. In addition, TransG uses a trick to control the parameter update process during training. For those triples that are very unlikely, the parameter updates will be skipped. Therefore, a condition similar to TransE is introduced in TransG, and the training algorithm updates the embedded word vector only if the following conditions are met:

$$\frac{\mathbb{P}\{(h,r,t)\}}{\mathbb{P}\{(h',r,t')\}} = \frac{\sum_{m=1}^{M_r} \pi_{r,m} e^{-\frac{\|\mathbf{u}_h + \mathbf{u}_r, m - \mathbf{u}_t\|_2^2}{\sigma_h^2 + \sigma_t^2}}}{\sum_{m=1}^{M_r} \pi_{r,m} e^{-\frac{\|\mathbf{u}_{h'} + \mathbf{u}_r, m - \mathbf{u}_{t'}\|_2^2}{\sigma_{h'}^2 + \sigma_{t'}^2}}} \leq M_r e^\gamma \quad (10)$$

where, $(h,r,t) \in \Delta$, $(h',r,t') \in \Delta'$, γ is learning rate.

Although this trick can shorten the learning time by skipping the impossible triples, for this article, as mentioned in the previous section, a large number of triples in the data set are skipped due to the self-sampling method, reducing the data usage. Therefore, we change the self-sampling method to adapt the algorithm to the purpose of this paper.

3.4 Opinion inference

After generating the knowledge graph consisting of attributes, opinions, and opinion words by the method shown in the previous section, the knowledge graph can be used to judge the opinions by the input (attribute, opinion word). The geometric meaning of the triple in TransG is expressed as Eq. (7), so it is easy to get the geometric meaning expression of opinion inference:

$$\mathbf{u}_{r,m} = \mathbf{t} - \mathbf{h} \quad (11)$$

In Eq. (5), a scoring function for the TransG has been given. In the opinion inference, it is only necessary to find the having the highest scoring function in the known word vector, namely:

$$\mathbf{u}_{r,m} = \arg \max_{r \in R, m=1 \dots M_r} \left(\pi_{r,m} e^{-\frac{\|\mathbf{u}_h + \mathbf{u}_{r,m} - \mathbf{u}_t\|_2^2}{\sigma_h^2 + \sigma_t^2}} \right) \quad (12)$$

All the elements in the Eq. (12) are known, so the opinion expressed by the (attribute, emotional words) can be inferred simply. For the knowledge base containing m attributes and n emotional words, $m > n$, the time complexity required by the method in this paper is only the time required to find attribute words and emotional words, and the time complexity of the algorithm is $O(m+n)$, for the traditional dictionary method, the whole dictionary needs to be searched to get the emotion expressed by the binary relation. According to the size of the dictionary, the time complexity of the algorithm is from $O(m)$ to $O(mn)$. Therefore, the method proposed in this paper has a significant improvement in computational efficiency compared to the traditional dictionary method except in the extreme case (minimum dictionary).

4 Results

The experimental environment of this paper is 64-bit Windows 10 OS, Intel Xeon E3 1230v2 processor, clocked at 3.30 GHz, 16 G memory, implementation language is Python, Python version 2.7, running environment PyCharm Community Edition 2016.

We crawled 12,902 pieces of data on the Pacific Auto Network. After the word segmentation and labeling method introduced in the previous chapter, we finally obtained 14,115 triples stored in (attribute, opinion, opinion word) as data set. Then the obtained data set is segmented, with 10,812 triples as the training set, 2,703 triples as the validation set, and 600 triples as the test set for subsequent training and testing. In order to more comprehensively test the accuracy of the model, this paper carried out a 10-fold cross-validation on the accuracy of the model, that is, the data set was divided into 10 parts, 9 of which are taken as training data and 1 part are used as test data for experiment. Each test will yield the corresponding accuracy, taking the average of the accuracy of the 10 results as an estimate of the accuracy of the algorithm. This method can reduce the specialty of the data set and carry out a more accurate evaluation. In this experiment, the triples of positive emotion and negative emotion were first divid-ed into 10 parts. Each test set was taken from the data sets of the two emotions to form a test set, and the rest was used as a training set. Since 14,115 (attribute, opinion, opinion word) triples cannot be evenly divided into 10 parts, the number of triples in the training set in the first 9 experiments is 12,704, and the number of test set is 1,411. In the last experiment, the number of triplets in the training set was 12,699, and the number of test set was 1,416.

Two tasks were used to test the accuracy of the different models, namely the opinion prediction task and the triple classification task. Among them, the opinion prediction task is to input the (attribute, opinion word) binary relation into the trained model, and predict the opinion in the triplet (attribute, opinion, opinion word), and HITS@1 is the probability that the correct opinion ranks the first. The triple classification task is to input a (attribute, opinion, opinion word), let the trained model calculate the relationship matrix

to determine whether the triple is the correct ternary group. In terms of parameters, TransE, TransH, and TransA all use the improved sampling method. The knowledge graph generated by the training is 50-dimensional, the learning rate λ of the model is 0.001, the training threshold γ is 1. The original TransG and TransG with improved sampling method share the same the parameters: the knowledge graph generated by the training has a dimension of 50 dimensions, the learning rate λ is 0.001, the training threshold γ is 3.5, and the CRP factor β is 0.025. The test results are shown in Tab. 1. Experiment result of different models Tab. 1.

Table 1: Experiment result of different models

Model	Opinion Prediction (HITS@1)	Triple Classification
TransE	87.1	79.8
TransH	88.8	79.8
TransA	88.8	75.1
TransG(Previous Self-Sampling)	90.8	79.6
TransG (Improved Self-Sampling)	92	85.3

As can be seen from Tab. 1, all the Translation models get good results in the opinion prediction task. Even using TransE, the simplest model, the result does not fall behind other models much. After analyzing, we tend to believe that the relationship is only ternary, the calculation complexity is not very high, so the disadvantage of simple model here is not fully demonstrated: in the TransE model, the distant between $\mathbf{l}_t - \mathbf{l}_h$ and the correct \mathbf{l}_r is far, while that between $\mathbf{l}_t - \mathbf{l}_h$ and the wrong $\mathbf{l}_{r'}$ is even further, so TransE will still choose the correct relationship according to the loss function. Compared with the TransE model, the algorithmic accuracy improvement of the TransH model and the TransA model is only 1.7%, limited, and the accuracy of the algorithm is challenged. The TransG model improves the accuracy of TransH and TransA by 2% by considering multiple semantic methods, higher than TransH and TransA to TransE. Among all models, TransG model has the highest accuracy, and in the generated $\mathbf{l}_t - \mathbf{l}_h$ vector diagram, TransG also shows a good clustering effect, as shown in Fig. 2.

In Fig. 2, the red point is the difference between the attribute vector and the opinion vector of the positive opinion, the gray point is that between the attribute vector and the opinion vector of the neutral opinion, and the blue point is that between the attribute vector and the opinion vector of the negative opinion. Although the vector graph generated by the TransG with original self-sampling in Fig. 2 has a certain clustering effect, the lattices generated by the three opinions are closely attached to each other and cannot be properly separated. In order to further improve the accuracy of the algorithm, we propose an improved sampling method. The vector graph after improving the self-sampling in TransG is shown in Fig. 3.

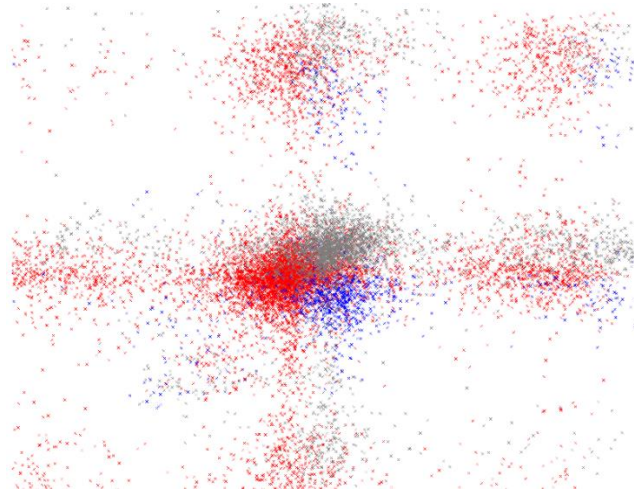


Figure 2: The vector graph generated by the original TransG

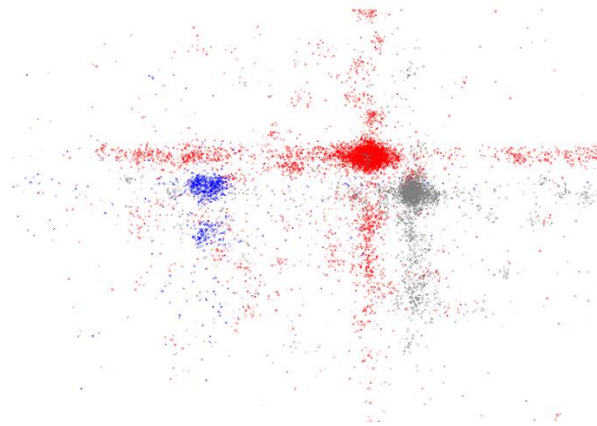


Figure 3: The vector factor generated by the improved TransG

By comparing Fig. 2 with Fig. 3, latter shows a significantly improvement in the clustering effect of the improved self-sampling method, and the scattered data is less, resulting in multiple dense lattices. Moreover, it can be clearly seen from the figure that both positive emotions and negative emotions have a larger and denser lattice, which can be inferred to be the semantics of the most commonly used (attribute, opinion word) collocation.

In Tab. 1, it can be seen that the improved self-sampling method has enhanced the accuracy by 1.2% compared with the original self-sampling method, indicating that the improved sampling method does have a certain effect. Although the accuracy difference is not that large, by analyzing the vector graph, it can be found that the improved sampling method will have a more obvious advantage when the data set becomes larger. Tab. 1 also shows that in the triple classification, the improved self-sampling method has increased from 79.6% to 85.3%, an increase of 5.7%, which is obvious.

Since the previous test was only performed on the 600-divided triples, there may be special cases where the data set is randomly categorized causing the high accuracy. In order to eliminate the specialty caused by the data set segmentation and prove the stability of the model proposed in this paper, the model is verified by 10-fold cross-validation. The results are shown in Tab. 2.

Table 2: 10-fold cross-validation result

No.	Training Set	Test Set	Opinion Prediction (HITS@1)%
1	12704	1411	90.4
2	12704	1411	91.9
3	12704	1411	90.9
4	12704	1411	90.1
5	12704	1411	90.6
6	12704	1411	91.2
7	12704	1411	92.3
8	12704	1411	90.6
9	12704	1411	93.3
10	12699	1416	90.4
Average			91.1

It can be seen that the average accuracy of the 10-fold cross-validation can reach 91.1%, which is similar to the 92% result in Tab. 2. It can be proved that the model has better accuracy and stability.

In order to test the speed of our algorithm, this paper randomly extracts 1000 (attribute, opinion word) from 14115 data sets and uses the traditional dictionary method and our method to predict the opinion. The time cost to discriminate the 1000 opinion that binary relation stands for is recorded separately. The results are shown in Tab. 3.

Table 3: Algorithm time consumption comparison

Method	Time
Dictionary	0.467 s
Our Method	0.429 s

As can be seen from Tab. 3, the method in this paper has a calculation speed of about 8% faster compared with the traditional dictionary method (the calculation of the promotion rate is (dictionary time-method time)/dictionary time). Since there are only 14,115 triples in the knowledge base, which number is not huge, the advantage of the calculation methods in Tab. 4 is not very obvious. Further research has found that in the face of collocations that do not exist in the dictionary, the dictionary method will take a lot of time and cannot produce results; and the method in the knowledge graph can still consume the same time in the table and get the correct result.

5 Conclusion

This paper designs the crawler script to obtain the evaluation statement on the car review website, and then establishes the rule template for word segmentation, extracts the (attribute, opinion word) binary relation and uses it as the smallest unit of emotion expression. Finally, the annotation of the experimental data set is completed by combining the existing emotion dictionary, the emotional information in the webpage and the manual labeling. The knowledge representation learning related knowledge and research are applied to the field of affective cognition, and the TransG is used to generate the knowledge graph and use it to complete the opinion discrimination. At the same time, the sampling method in the original model is improved, so that the data set obtained in this paper can be more fully utilized and the clustering ability of the model is improved. At present, the ternary emotion is judged, but the emotions in real life are much more complicated than the grading of emotional intensity, such as positive, negative and neutral. Therefore, increasing the category of emotional judgment is an important work to follow, and it is worthy of further study.

Acknowledgement: This research is supported by the Key Program of National Natural Science Foundation of China (Grant Nos. U1536201 and U1405254), the National Natural Science Foundation of China (Grant No. 61472092).

References

- Blair-Goldensohn, S.; Hannan, K.; McDonald, R.; Neylon, T.; Reis, G. A. et al.** (2008): Building a sentiment summarizer for local service reviews. *WWW Workshop on NLP in the Information Explosion Era*, vol. 14, pp. 339-348.
- Bordes, A.; Usunier, N.; Garcia-Duran, A.; Weston, J.; Yakhnenko, O.** (2013): Translating embeddings for modeling multi-relational data. *Advances in Neural Information Processing Systems*, pp. 2787-2795.
- Brody, S.; Elhadad, N.** (2010): An unsupervised aspect-sentiment model for online reviews. *Human Language Technologies: the 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pp. 804-812.
- Hatzivassiloglou, V.; McKeown, K. R.** (1997): Predicting the semantic orientation of adjectives. *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics*, pp. 174-181.
- Ji, G.; He, S.; Xu, L.; Liu, K.; Zhao, J.** (2015): Knowledge graph embedding via dynamic mapping matrix. *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, vol. 1, pp. 687-696.
- Jin, W.; Ho, H. H.; Srihari, R. K.** (2009): OpinionMiner: a novel machine learning system for web opinion mining and extraction. *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1195-1204.
- KIM, S. M.; Hovy E.** (2004): Determining the sentiment of opinions. *Proceedings of the 20th International Conference on Computational Linguistics*, pp. 1367-1343

- Lin, Y.; Liu, Z.; Sun, M.; Liu, Y.; Zhu, X.** (2015): Learning entity and relation embeddings for knowledge graph completion. *Twenty-Ninth AAAI Conference on Artificial Intelligence*, vol. 15, pp. 2181-2187.
- Liu, B.; Zhang, L.** (2012): A survey of opinion mining and sentiment analysis. *Mining Text Data*, pp. 415-463.
- Liu, Y.; Chen, Y.** (2015): Research on Chinese micro-blog sentiment analysis based on deep learning. *8th International Symposium on Computational Intelligence and Design*, vol. 1, pp. 358-361.
- Mikolov, T.; Chen, K.; Corrado, G.; Dean, J.** (2013): Efficient estimation of word representations in vector space. arXiv:1301.3781.
- Socher, R.; Perelygin, A.; Wu, J.; Chuang, J.; Manning, C. D. et al.** (2013): Recursive deep models for semantic compositionality over a sentiment treebank. *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pp. 1631-1642.
- Su, Q.; Xu, X.; Guo, H.; Guo, Z.; Wu, X. et al.** (2008): Hidden sentiment association in chinese web opinion mining. *Proceedings of the 17th International Conference on World Wide Web*, pp. 959-968.
- Turian, J.; Ratinov, L.; Bengio, Y.** (2010): Word representations: a simple and general method for semi-supervised learning. *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pp. 384-394.
- Wang, Z.; Zhang, J.; Feng, J.; Chen, Z.** (2014): Knowledge graph embedding by translating on hyperplanes. *Twenty-Eighth AAAI Conference on Artificial Intelligence*, vol. 14, pp. 1112-1119.
- Xiao, H.; Huang, M.; Hao, Y.; Zhu, X.** (2015): TransA: an adaptive approach for knowledge graph embedding. arXiv preprint arXiv:1509.05490.
- Xiao, H.; Huang, M.; Zhu, X.** (2016): TransG: A generative model for knowledge graph embedding. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, vol. 1, pp. 2316-2325.
- Yang, M.; Tu, W.; Wang, J.; Xu, F.; Chen, X.** (2017): Attention based LSTM for target dependent sentiment classification. *Thirty-First AAAI Conference on Artificial Intelligence*, vol. 17, pp. 5013-5014.