Research on Semantic Prediction Analysis of Tibetan Text Based on Word2Vec

Ding Hai-lan¹a, Yu hong-zhi²b, Qi kun-yu³c

¹Northwest Minzu University, China National Institute of Information Technology, Gansu Lanzhou 730030,
²Northwest Minzu University ,China National Institute of Information Technology, Gansu Lanzhou 730030,
³Northwest Minzu University ,China National Institute of Information Technology, Gansu Lanzhou 730030.

a-e-mail: www.dinghailan@qq.com
b-e-mail: www.zwbgzdh@126.com
c-e-mail: www.2630103542@qq.com

Abstract. This article uses Google's open source Word2Vec tool to input the corpus of the Tibetan text "Sage wedding" after the word segmentation. The words are mapped to a K-dimensional space in the text and transformed into word vectors by using the context information of the vocabulary. The Word2Vec tool then learns to get a vector model, each of which is represented by a unique word vector. A vocabulary is constructed through training text data and then the vector is represented by learning the words. Word vectors capture the laws of many languages, which results the similarity of the distance between words and words. The experimental results show that the accuracy and recall rate based on the Word2Vec training model are very high.

1. Introduction

In 2013, Tomáš Mikolov and his colleagues at Google released Word2Vec, which is an unsupervised tool for computing continuous distributed representations of words. The tool provides a vector representation of the continuous word bag model CBOW and Skip-Gram models for calculating words. The Word2Vec tool requires a text corpus as input and produces a word vector as output. Firstly, it constructs a vocabulary based on the training text data and then learns the vector representation of the word. The generated word vector file can be used as a function in many natural language processing and machine learning applications. This tool finds the closest word for a user-specified word. For example, if you enter "China", the distance will display the most similar words and their related words that are closest to the "China" distance. In the article, the Tibetan text is taken as an example of the "Sage wedding " because it is a profound Tibetan historical work. Its writing is exquisite, including Tibet's history, religious history, cultural development and the economy of ancient Tibet. It has a unique style historic value. In this paper, Word2vec's CBOW training model is used to express words into word vectors quickly and efficiently. The word vectors capture the semantic features between words in natural language, and then output according to distance and distance to output the most
similar or related to high frequency words. Vocabulary, these words serve as important information to predict the approximate semantics of text.

2. **Word2 Vector**

The computer can only recognize 1 and 0 for natural language processing, then if you want the computer to process the text, you must convert the text into a language that the computer can recognize. The most direct way is to convert the word into a word vector. The vectorization of words refers to the mathematical representation of words, mainly including one-hot representation, distributed representation and word2vec model training word vector. The first way is to use a very long vector to represent a word with a component of 1, and the rest are all 0, 1. The shortcoming is that it cannot provide semantic information. The second way was first proposed by Hinton, who mapped the words into a low-dimensional and dense 100-200-sized real vector space, so that the closer the words are, the closer the word distance is. The third way is to use the NNLM model (Nin Network Language Model) proposed by Bengio and the Log-Linear model and the Mikolov model of Hinton to propose the language model of Word2Vec. Word2vec can effectively train the word vector at high speed. [1]

3. **Word2 Vec Tool**

Word2Vec is a vector model representation tool. The tool turns words into vectors, and calculates the similarity by calculating the cosine between the two word vectors. The Word2Vec tool takes the corpus after the word segmentation as input and learns to get a vector model. It first constructs a vocabulary from the training text data and then learns the vector representation of the word. Word vectors can capture the laws of many languages, such as the operation of a vector: vector ("ri bo"), vector ("ri mthon po"), vector ("ri khul"), vector ("ri rts")etc, translated into “mountain, mountain, mountain, mountain top” in order. Through the context operation, the semantic distance between the vector and the vector is very similar in the results obtained by the above examples. Therefore, we can understand Word2Vec as a distance tool. When you input a word, it will display all words close to the word according to the distance.

3.1. **Two training models of Word2Vec**

The word vector is actually a vector that maps words to a semantic space. Word2Vec is implemented by means of neural network. Considering the context of text, Word2Vec has two models, CBOW and Skip-gram. These two models are similar in the training process. The Skip-gram model uses a word as input to predict the context around it. The CBOW model takes the context of a word as input to predict the word itself.

3.1.1. Skip-gram training model

If you use a word as input to predict the context around it, the model is called the Skip-gram model. First determine the window size Window, generate 2*window training samples for each word, (i, i-window), (i, i-window+1),..., (i, i+window-1), (i, i+window). Immediately after determining batch_size, note that the size of batch_size must be an integer multiple of 2*window, which ensures that each batch contains all samples corresponding to one vocabulary. There are two training algorithms: Hierarchical Softmax and Negative Sampling. [2] Finally, the neural network is iterated and trained for a certain number of times, and the parameter matrix of the input layer to the hidden layer is obtained. The transposition of each row in the matrix is the word vector of the corresponding word. The specific model is shown in Figure 1:
3.1.2. CBOW training model

The CBOW (Bag-of-words model) model takes the context of a word as input to predict the word itself. Firstly, it determines the window size window, generate 2*window training samples for each word, (i-window, i), (i-window+1, i), ..., (i+window-1, i), (i+window, i). Secondly, it determines the batch_size. Note that the size of the batch_size must be an integer multiple of 2*window. This ensures that each batch contains all the samples corresponding to one vocabulary. There are two training algorithms: Hierarchical Softmax and Negative Sampling. Finally, the neural network is iterated and trained for a certain number of times, and the parameter matrix of the input layer to the hidden layer is obtained. The transposition of each row in the matrix is the word vector of the corresponding word. The specific model is shown in Figure 2:

In the two training models of Word2Vec's Skip-gram and CBOW, it is recommended to use the Skip-gram training model to train when there are more corpora of training. When the corpus is relatively small, it is recommended to use the CBOW training model to train. In general, Word2Vec can use the trained word vector model, convert the input words into word vectors and then train the model, and finally output the words according to the distance, and turn these words into a synonym set.

4. Experimental process and results analysis

First, we select the Tibetan text corpus "Sage wedding" (mkhsa p’i dg’a ston ) which size is 2.37MB and import the text into the Tibetan WordSegmentEx3.0 Tibetan word segmentation software, and get the text after the word segmentation and the text sorted according to the word frequency. Next, we install Anaconda3, an open source Python distribution, which is an open source package management system and environment management system. After the installation is successful, use the Conda to build a development environment in the terminal.

4.1. Experimental steps

After the terminal builds the environment variables, in the Spyder development environment of Anaconda3, the word vector test code is written in the python programming language. We call the Word2Vec CBOW model algorithm in the Gensim toolkit to train. The size of the trained word vector is 50, the training window is 5, and the minimum word frequency is 5. Firstly, we calculate the similarity of two words, then calculate the related words of a word, and finally output the set of words that are closest to the semantic distance of the two words. The core code for a word vector test written in the python programming language is as follows:

```python
# genism modules
from genism.models import Word2Vec
from genism.models.word2vec import Text8Corpus
import os.path
```
import sys
import numpy as np

The corpus of training is after the word segmentation, the batch processing has 33244 sentences for each sentence. Then, all the part-of-speech annotations in the corpus are removed and the tokens are replaced by spaces. Each sentence retains the ending symbol in the Tibetan sentence, that is, the single suffix, which is the unit of the sentence. Finally, we get a list of features for a text. In the word bag model (CBOW), the characteristic of a document is the word it contains. The specific steps are shown in Figure 3:

![Diagram](image)

Figure 3 Word2Vec experimental steps

After extracting the feature table, the high-frequency vocabulary is selected for training. When the high-frequency vocabulary is selected, the helper words [3] (rnama dbye), conjunctions (dnga sgra), numerals (grngs p'I tshig), decorative set words (rgyna sdud), pronouns (tshba tshig), to-be-reported words (lhga bcsa), negative words (dgga sgra), refers to human nouns suffix (bdga sgra), adjectives (khyda chos kyi ming) and single vertical symbols (shda) in the word frequency table are removed[4]. After removing these high-frequency vocabulary and symbols, select 10 sets of high-frequency vocabulary for training. The specific word frequency is shown in Table 1:

| High frequency word | Word frequency | High frequency word | Word frequency | High frequency word | Word frequency |
|---------------------|----------------|---------------------|----------------|---------------------|----------------|
| Rgyla po            | 1035           | chos                | 917            | mdada               | 747            |
| byung               | 731            | lo                  | 650            | bysa                | 640            |
| Sngsa               | 474            | bod                 | 470            | byon                | 410            |
| Bstna pa            | 406            | shing               | 399            | dus                 | 395            |
| bya                 | 380            | gnsa                | 374            | Ston pa             | 353            |
|                     |                |                     |                |                     | 346            |

10 groups of high frequency words are used as the training target to enter the Word2Vec model for training. The training results are shown in Table 2, Table 3, Table 4 and Table 5:

| High frequency word | Word frequency | High frequency word | Word frequency | High frequency word | Word frequency |
|---------------------|----------------|---------------------|----------------|---------------------|----------------|
| rgya po             | 0.8170989751815796 | chos               | 0.681171059684595 | mdada               | 0.8219879865646362 |
| gsms ngga           | 0.8086523413658142 | bynga phsys        | 0.6710954112052917 | ion pa              | 0.7987777590751648 |
| sgrub               | 0.807623922828596 | shing rta          | 0.6692179441452026 | rgyla sa            | 0.789211094379425 |
| bsgrubs             | 0.8062039017673307 | zha snga           | 0.6634161472320557 | bshe pa             | 0.779091596633936 |
| thugs               | 0.8038085690901421 | sbye'u             | 0.6567350625991821 | bsheos              | 0.7712806463241577 |
| dge b'i lhase        | 0.7964973449707031 | mdog               | 0.656122684787598  | druk bcu            | 0.7650943994522095 |
In the development environment of the Word2Vec model, when the two words of the highest word frequency "Rgyla po chos" (king, religion) are input in the text training, the words that are closest to and related to the two words are obtained in turn "Spel, gdms ngga, sgrub, bsgrubs, thugs, dge b'i bshes gnyen, rje 'bngsa, gnda, bon, mna ngga" (development, teaching, practice, achievement, spirit, monarch, elite, Bon and teaching). When you enter the two terms of "Mdad tshe" (career, time), you will get the words that are closest to and related to the two words "bynga phyoks, phying, shing rta, zhla snga, khye'u, mdog, dbus ma, phok pa, rjes su bzung, smin mtshma" (North, Backward, Big Car, Pre-driver, Junior, Color, Center, Teaching, Support and Eyebrow). From the above high-frequency vocabulary set, we can see that the relevant words obtained by training and the close-range words are all related to the Buddhist language, the development of Buddhism, and the historical part of the evolution of the word, so that we can roughly predict the semantics of the text is about religion which is a work of history.

Table 3 Word2Vec model training related words of two words

| sngsa rgya | bod | bya | lha | byon | byed |
|------------|-----|-----|-----|------|------|
| rgya gra   | 0.8007855415344238 | 0.7499148845672607 | 0.7952665866073611 |
| dra ba     | 0.7687294475555554 | 0.7326218488532214 | 0.7951421164244511 |
| mu stegs   | 0.7643926143646244 | 0.719208961601257 | 0.7947723330808111 |
| rgya khmsa | 0.7598621748973145 | 0.7027950286865624 | 0.7946202518927917 |
| jo bo rje  | 0.7370045185089111 | 0.7001242637634277 | 0.7859824895858765 |
| gdong dmra | 0.729033873836322 | 0.699469835777283 | 0.7819713354107181 |
| srol       | 0.7245049941619589 | 0.697236104966521 | 0.7802630662918091 |
| bon        | 0.7234969139099912 | 0.6975480318069458 | 0.7792367307920325 |
| shra phyogs| 0.7211406239041201 | 0.6925481494108887 | 0.7787083983421326 |
| bla        | 0.7209500074386597 | 0.6915058493614197 | 0.7785230875015259 |

In the development environment of the Word2Vec model, when the two words of the word frequency"sngsa rgya bod" (Buddha, Tibetan) are input in the text, the words that are closest to and related to the two words are obtained in turn "rgya gra, dra ba, mu stegs, rgya khmsa, jo bo rje, gdong dmra, srol, bon, shra phyogs, bla" (India, development, foreign roads, national land, Jue Wojie, red face, Customs, Bon, Eastern and Nepal). From the above phrases, we can also clearly see that the phrases obtained from the training are all about the common language in Tibetan Buddhism, such as: "The Emperor Shi Tian, Chi Zun and the Living". Secondly, through training, we can know that the text mainly describes the relationship between Tibetan Buddhism and Nepal and India. Furthermore, we can see the expression of Buddhist culture in the Tibetan region through the word frequency: "Musical instruments, wine, sound, playing and goddess". In general, we can predict the semantics of the text about the prevalence of Buddhism in Tibet and the development of culture.

Table 4 Word2Vec model training related words of two words

| brtua pa | shing | bya | lla | byon | byed |
|----------|-------|-----|-----|------|------|
| bcsa  | 0.7470698164978027 | 0.8041076719459534 | 0.8767780655634699 |
| phy nga | 0.742723988723755 | 0.758179664618164 | 0.8544735442514651 |
| lha 'dre | 0.7472052435874939 | 0.7580803592491149 | 0.81573190626773 |
| ldn pa  | 0.7539792527961731 | 0.7578626527159033 | 0.84351314146626 |
| sgra 'byin | 0.7373657822888948 | 0.743770787191667 | 0.8402106168585925 |
| te    | 0.7352505265750107 | 0.7417047023771393 | 0.8392305374145508 |
| phba pa | 0.7346739768919349 | 0.7383226156247341 | 0.8358529840469316 |
In the development environment of the Word2Vec model, when the two words of the word frequency “bstna pa shing” (religion, tree) are input in the text, the words that are closest to and related to the two words are obtained in turn “bcsa,Phyi nnga,Lha'dre,dun pa ,Sgra 'byin,te,Phba pa,SKU gsung thugs rten,Gso ba,Sa rdo” (with, inside, outside, ghosts, Desire, emperor's interpretation, words to be preached (ospace)), distiller's yeast, Buddha statue, education and stone). When the two words of the word frequency “dus bu” (time, man) are input in the text, the words that are closest to and related to the two words are obtained in turn “Bu mo,Srsa mo,Nka mo,nya,lei,Che ba,za,giu,rba,Gitsug lg” (girl (honorific), Saturn, Sun, heavy, big, food, jasper, wave and holy book). When the two words of the word frequency "bya knda" (bird, place) are input in the text, the words that are closest to and related to the two words are obtained in turn “Gnsa pa, Bye ba,Bsma gtna,Rnma pra,Gling bzhi,Zhing khmsa,Mth'a,Thun mong,Stug po,Chos kyi rnm krngsa” (existence, tens of thousands, meditation, state, four continents, heaven, margin, common, dense and methodic). From the above related phrases of phrases and phrases, I can clearly see the vocabulary inside Buddhism. It is not common words that have comments to understand, but we can confirm that the text is still about the history and culture of Buddhism. Text semantics can also predict text semantics through high-frequency words and related words that are closest to each other.

| Ston pa jul | Grong khyer | Ston pa jul | Grong khyer |
|------------|-------------|------------|-------------|
| Za hor     | 0.7662957906723022 | Rgya khmsa | 0.752292754287712 |
| Bynga phyogs | 0.7599533200263977 | Ao rgyna | 0.7498317956924438 |
| Nub phyogs | 0.7539993524551392 | Brgya byin | 0.7400675415992737 |
| Chos kyi rnm | 0.8344778735923767 |
| Gbps       | 0.8326832056045532 |

In the development environment of the Word2Vec model, when the two words of the word frequency “Ston pa jul” (master, mountain god) are input in the text training, the words that are closest to and related to the two words are obtained in turn “Za hor,Bynga phyogs,Nub phyogs,Rgya khmsa,Ao rgyna,Brgya byin,Rkya nga,Shra phyogs,Gdong dmra,Grong khyer” (Ancient Little Kingdom of India, North, West, Land, carefulness, village, country, east, red face and city). From the last set of phrases and the most recent and related phrases predicted, the text is a unique historical value that describes the relationship between ancient India and Tibetan Buddhism and the economy of ancient Tibet.

5. Conclusion
This paper uses the open source Word2Vec tool of GOOGLE to input Tibetan text as corpus, and transforms the word in the text into the word vector by using the vocabulary context information. By using the CBOW model algorithm model in Word2Vec to train the laws of many languages, the distance between words is the similarity. Further, through the high-frequency vocabulary as input, the vocabulary closest to the high-frequency vocabulary can be output through training, and the high-frequency words and the similar vocabulary are used as important information to predict the semantics of the text. This method plays a fast and convenient role in quickly grasping the subject semantics in the long corpus. At the same time, many interesting language rules can be found through training, which avoids the problem of long duration and subjective judgment. However, many words found in the training did not appear in the context, which brought some errors to the semantic prediction. In general, Word2Vec tools can effectively predict text semantics.

Acknowledgment
2018, Northwestern University for Nationalities Graduate Research (Practice) Innovation Project "Word2vec-based vocabulary clustering research in Tibetan field (Yxm2018011)"); National Science and Technology Fund Project "Research on Literary Cognition Theory of Tibetan Lhasa Based on
EEG Signals" (61462075) "; Gansu Province Innovation Group Project: Tibetan Intelligent Information Processing (Z16088).

**About the Author**
Ding Hailan (1990-), female (Tibetan), Tianzhu, Gansu, Ph.D., mainly engaged in Tibetan information processing and cognitive linguistics research.

**References**
[1] Tang Ming, Zhu Lei, Zou Xianchun. A Document Vector Representation Based on Word2Vec. [J] COMPUTER SCIENCE computer. 2016.06.
[2] Xie Rimin, Chen Jie, You Guirong, Xie Datong. Research on Chinese Book Classification Based on Word2Vec. [J] Yunnan Nationalities University. 2018.07.
[3] Ding Hailan, Qi Kunyu. A Comparative Study of Tibetan Texts and English Prepositions. [J] Linguistics. 2016.3.
[4] Ma Jinwu. Four clear structures of Tibetan grammar. [M]. Beijing Ethnic Press.2008.9.