WEP Warwick Evans Publishing

Transactions on Economics, Business and Management Research
ISSN: 2960-1789, eISSN: 2960-2254 | Volume 2
MEHSS 2023

Tibetan Syllables Similarity Calculation Based on Word2Vec

Yanhua Duan [a], Xianghe Meng [b, *]

Key Laboratory of Linguistic and Cultural Computing Ministry of Education, Northwest Minzu University, Lanzhou 730030, China

[a] 446549989@qq.com, [b, *] 554597360@qq.com

* Corresponding author: Xianghe Meng

Abstract. This article focuses on the vector representation of Tibetan syllables as the basic text representation unit, and uses the CBOW model in Word2Vec as the representation method for syllable vectors. In terms of calculating the similarity between Tibetan syllables, different window sizes are set to obtain semantic relatedness information at different levels of syllables, and the average similarity between syllables is determined based on the similarity results calculated by multiple models. Finally, three examples of the performance of syllable similarity between three syllables are presented to demonstrate the feasibility of using syllable similarity calculation for syllable correction.

Keywords: Tibetan Syllables; Similarity Calculation; Word2Vec; Syllable Vectors.

1. Introduction

Tibetan belongs to the Tibetan-Burmese branch of the Sino-Tibetan language family within the Tibetic languages. In China, the language is mainly spoken in the Tibet Autonomous Region, Ganzi Tibetan Autonomous Prefecture, Aba Tibetan-Qiang Autonomous Prefecture, Gannan Tibetan Autonomous Prefecture, and Diqing Tibetan Autonomous Prefecture. Outside of China, Tibetan is spoken in countries such as Bhutan, India, Nepal, and Pakistan[1]. Tibetan syllables are composed of 30 consonant characters and 4 vowel characters that are combined in a vertical and horizontal manner, with a core character as the base and arranged in a vertical and horizontal spelling system[2]. Tibetan is a written language composed of Tibetan letters, which have both phonetic and semantic features. Tibetan is a complex writing system with both phonetic and semantic features due to its belonging to the Sino-Tibetan language family[3]. Modern Tibetan grammar, also known as Tibetan grammar, was established after the third revision[4]. Written Tibetan that conforms to modern Tibetan grammar is called modern Tibetan[5]. In this article, Tibetan refers to modern Tibetan.

The structure of Tibetan syllables is very complex. The specific positions of Tibetan characters in syllables can be called "construction positions". According to Tibetan grammar, there are certain restrictions on the characters, properties, and quantities that appear in each construction position of Tibetan syllables, and there are also constraint relationships between them. Each construction position can be represented in Tibetan syllables as: the preceding addition position, the upper addition position, the base character position, the lower addition position, the vowel position, the rear addition position, and the further rear addition position, which correspond to the preceding addition character(ཀ,ད,བ,མ,འ),the upper addition character(ར,ལ,ས),the base character (ཀ,ཁ,ག,ང,ཅ,ཆ,ཇ,ཉ,ཏ,ཐ,ད,ན,པ,ཕ,བ,མ,ཙ,ཚ,ཛ,ཝ,ཞ,ཟ,འ,ཡ,ར,ལ,ཤ,ས,ཧ,ཨ), the lower addition character (ཡ,ཡ,ར,ལ), the vowel character (ི, ུ, ེ, ོ), the rear addition character (ག,ང,ད,ན,བ,མ,འ,ར,ལ,ས), the further rear addition character (ད,ས).

In 1995, Jiang Di et al.[6] referred to the 30 consonant letters and 4 vowel letters in Tibetan as Tibetan characters. They believed that each Tibetan string that forms a single syllable may represent a word or a morpheme in Tibetan language. Tibetan phonetics is monosyllabic, and each Tibetan string that forms a syllable represents a syllable in Tibetan language. The combination of each Tibetan character is called a Tibetan word. The modern Tibetan grammar imposes certain restrictions on the characters,

properties, and quantities that can appear in each construction position within Tibetan syllables, and there is also a constraint relationship between the construction positions. Tibetan syllables contain a lot of information, such as tense, vocabulary, and grammar[7]. Kong Jiangping et al.[8] investigated more than 6,000 Tibetan syllables from the perspective of ancient Tibetan initials and finals in the book "Survey of Tibetan Dialects", which helps readers understand the sound system of ancient Tibetan. Wang Fuzhao et al.[9] conducted a study on error detection methods in Tibetan syllables, using a method based on the constraint matching between character components recognized by Tibetan character recognition and direct matching of syllables in the syllable library to detect errors in Tibetan syllables.

Cai Zhijie et al.[2] conducted a study on the model of modern Tibetan character attribute analysis system by analyzing Tibetan language corpus statistics and the structure of modern Tibetan characters. They designed a basic component character table library, a combined component character table library, a coarse-grained structural character table library, and a fine-grained structural character table library, and explained the structural characteristics of each character table library and introduced the Tibetan character attribute analysis algorithm. Zhu Jie et al.[10] proposed the TSRM Tibetan syllable spelling check algorithm, which focuses on modern Tibetan syllables. Based on the rules of modern Tibetan grammar, the algorithm establishes a Tibetan syllable rule model, which divides Tibetan syllables into prefixes (containing 224 elements), vowels (containing 5 elements), and suffixes (containing 18 elements). Duola et al.[11] conducted a study on syllable frequency of use based on Tibetan dictionaries, Tibetan language textbooks, epics, and corpora covering humanities and natural sciences. Liu Huidan et al.[12] conducted a statistical analysis of Tibetan syllable spelling errors based on a large-scale Tibetan network corpus and identified four main reasons for spelling errors: firstly, excessive use of vowel symbols; secondly, missing syllable nodes or sentence-ending spaces; thirdly, multiple expressions of the same Tibetan character or character; and fourthly, incorrect use of similar characters.

2. Tibetan Text Corpus and Representation of Syllables Vectors

2.1. Construction of Tibetan Text Corpus

Table 1. Information on the Source of the Tibetan Text Corpus.

Website Name	Website URL	Number of Texts (in 10,000s)
Qinghai News Network	https://www.qhtibetan.com/	6.81
China Tibet Online (Tibetan version)	https://ti.tibet3.com/	3.97
Qinghai Tibetan Network Radio and Television Station	http://www.qhtb.cn/	1.53
Tibet Daily Tibetan Edition	http://epaper.chinatibetnews.com/xzrbzw/	1.25
China Tibet Voice (Tibetan version)	http://www.vtibet.cn/	0.93
People's Daily Tibetan Edition	http://tibet.cpc.people.com.cn/	0.61
Tibet News Network	http://xizang.news.cn/	0.40

The corpus used in this article comes from Tibetan online news texts. News texts have the characteristic of relatively standardized vocabulary and usage, which can reflect the most commonly used language usage. The main websites providing the corpus in this article include China Tibet Online, Qinghai Tibetan Network Radio and Television Station, Tibet Daily Tibetan Edition, and

Qinghai News Network, totaling 7 websites with a total corpus size of 155,000 articles. Please refer to Table 1 for the information on the source of the text corpus. The publication years of the news texts mainly fall between 2012 and 2022.

2.2. Representation of Tibetan Syllables Vectors

Word vector representation is the optimal way of representing linguistic text units as vectors, enabling computers to better understand natural language. In natural language processing tasks, since machines cannot directly understand human language, the first thing that needs to be done is to digitize language, and word vectors are a good solution for this. Word vectors are not only applicable to "words". In fact, more granular or coarse-grained language units can be used for further research, such as character vectors[13], sentence vectors, or document vectors[14], which can achieve better representation of characters, sentences, or documents. This article conducts research on syllable vector representation using Tibetan syllables (similar to Chinese characters) as the basic text representation unit. The main application directions of Tibetan syllable vector representation are: first, using the trained syllable vectors as input features to complete natural language processing related tasks, such as being used as the input layer of machine learning models for text classification, sentiment analysis, and machine translation; second, from the perspective of linguistics, applying syllable vectors, such as using the distance between syllable vectors to represent the similarity between syllables.

Generally, word vector representation techniques can be divided into two categories: traditional word vector representation methods and neural network language model-based word vector representation methods. Traditional word vector representation methods mainly refer to using one-hot encoding to represent word vectors. In contrast, neural network language model-based word vector representation methods use word vectors as auxiliary parameters to help construct the objective function of the language model. During the training process of a neural probabilistic language model, word vectors can be seen as a byproduct of generating the language model. In fact, the language model and word vectors are bound together, and when the model training is completed, both are obtained simultaneously.

Neural network language models do not calculate conditional probabilities through counting, but instead build models through neural network structures. Compared with n-gram language models, the advantage of neural probabilistic language models is that the similarity between words can be calculated using word vectors. For example, in the Tibetan language corpus, sentence s_1 = "ཁྱི་ནི་སྡུག་རྩེ་བའི་སྐྱེས་ཕྱུག་རིན།" (Dogs are cute animals) appears 2000 times, while sentence s_2 = "རྟ་ནི་སྡུག་རྩེ་བའི་སྐྱེས་ཕྱུག་རིན།" (Horses are cute animals) appears only twice. According to the n-gram language model, $P(s_1)$ is expected to be much greater than $P(s_2)$. It can be seen intuitively that the only difference between the two sentences is the subject words "ཁྱི" (dog) and "རྟ" (horse), while the rest of the sentence content is exactly the same. Therefore, $P(s_1)$ and $P(s_2)$ should be close to each other to be correct. The results obtained by using a neural network language model to calculate are roughly equal for two main reasons. First, the neural probabilistic language model assumes that similar words correspond to similar word vectors. Second, the probability function is smooth with respect to word vectors, so when a component in the word vector changes slightly, the impact on the probability function is also small.

Using neural network language models to represent word vectors in a corpus has become the main method, and common methods include Word2Vec [15], BERT [16], etc. Word2Vec was proposed by Mikolov et al. in 2013 and employs two models, CBOW and Skip-Gram, both of which are based on neural network language models. To improve the performance of the model, the hidden layer and word order information in the original neural network language model are removed, and a new network model is obtained for generating word vectors. The word vectors of the context words in the neural network language model are represented by the average of the context words. Assuming that a language text sequence is "……, $word_{i-2}, word_{i-1}, word_i, word_{i+1}, word_{i+2}$,……", the CBOW model of Word2Vec predicts the target word by the occurrence of the context words around the target

word. As shown in the left example in Figure 1, the target word ($word_i$)is predicted based on the context words ($word_{i-2}, word_{i-1}$, $word_{i+1}$ and $word_{i+2}$). The Skip-Gram model predicts the context words of a given target word. As shown in the right example in Figure 1, the context words ($word_{i-2}, word_{i-1}, word_{i+1}$ and $word_{i+2}$) are predicted based on the given target word ($word_i$).

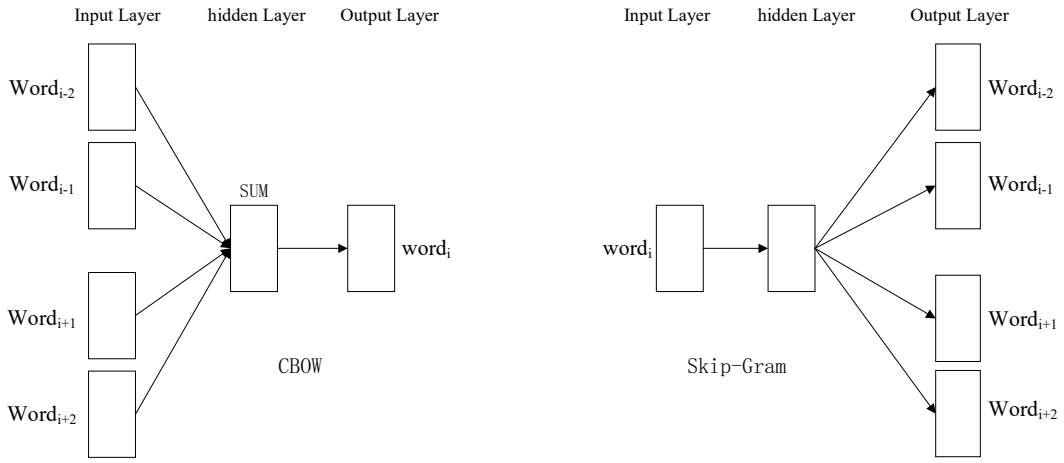

Figure 1. The two training models of Word2Vec

In this article, the principle for selecting a word vector representation method was that the vector representation should meet the basic experimental requirements, have good performance, and be practical for actual operations. Due to the need for large amounts of computational resources for excellent pre-trained models like BERT, which the laboratory does not have the basic conditions to conduct such research, this article selected the Word2Vec method for word vector representation. Regarding the selection of the two models in the Word2Vec method, CBOW and Skip-Gram, the experiment referred to the results of Cai Zhijie's paper "Construction of Tibetan Word Vector Similarity and Correlation Evaluation Sets".[17] Cai Zhijie conducted extensive experiments on the influence of CBOW and Skip-Gram models on Tibetan word vector representation from aspects such as learning rate, window size, vector dimension, and iteration times. The experiment showed that the CBOW model trained on Tibetan word vectors had better performance than the Skip-Gram model. Therefore, this article chose the CBOW model in Word2Vec as the final word vector representation method.

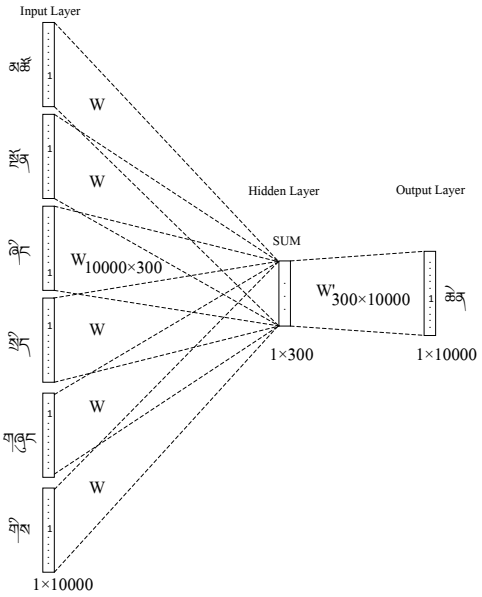

Figure 2. CBOW Model Workflow

Tibetan syllables are used as the basic language unit for syllable vector representation, and the workflow of the CBOW model is explained in detail. The example Tibetan sentence used is "མཚོ་སྔོན་ཞིང་ཆེན་སེམས་ཁྱབ་གཞུང་གིས་རྒྱུན་ལས་སྤྱིས་ཚོགས་བསྡུས།" (Qinghai Provincial Government held a routine meeting). First, the text is segmented into the format required by the model using syllable segmentation: "མཚོ སྔོན ཞིང ཆེན སེམས ཁྱབ གཞུང གིས རྒྱུན ལས སྤྱིས ཚོགས བསྡུས". The target syllable is set to "ཆེན", and the model window is set to 7. The CBOW model needs to predict the target syllable "ཆེན" based on the previous 3 syllables ("མཚོ", "སྔོན", "ཞིང") and the following 3 syllables ("སེམས", "ཁྱབ", "གིས") of the target syllable.

Suppose the number of syllable types in a Tibetan text corpus is 10,000, the dimension of the hidden layer is 300, and the input weight matrix is W (the size of this matrix is 10,000 * 300), and the matrix initialization has been completed. The output weight matrix is W' (the size of this matrix is 300 * 10,000), and the matrix initialization has also been completed.

Taking the example data into consideration, Figure 2 shows the specific workflow of the CBOW model, and the detailed steps are as follows:

Step 1: Based on the number of syllable types in the corpus, generate one-hot encoding for the six syllables of the target syllable context, that is, represent each syllable as a "1*10000" vector, which serves as the input layer input value.

Step 2: Multiply each of the six one-hot vectors from step 1 by the shared input weight matrix W, then add them together and take the average to obtain the hidden layer vector (size 1*300).

Step 3: Multiply the hidden layer vector by the output weight matrix W, then apply an activation function to obtain a probability distribution of 10,000 dimensions. The index with the highest probability corresponds to the predicted syllable. Compare the predicted syllable with the one-hot vector of the target syllable, and the smaller the error, the better. Use the cross-entropy loss function and gradient descent algorithm to update W and W'. After training the model, the one-hot vector of each input syllable in the input layer multiplied by matrix W is the corresponding word vector. Table 2 shows an example of the 300-dimensional vector information of the six input syllables obtained after training on large-scale corpus data (due to the size of the table, the vector values in the middle of the syllables are omitted and represented by "...").

Table 2. Example of a 300-dimensional vector for the six context syllables

Tibetan Syllables	Syllables Vector
མཚོ	[-1.8697063, 3.463201, -1.121849, ……, 1.8224212, 0.58887583, -2.6085038]
སྔོན	[2.5722482, -0.365581, 3.361344, ……, -1.4226035, -1.1193558, -0.20934965]
ཞིང	[-2.8141584, 1.7882948, -1.2218108, ……, -1.8009993, -2.4461539, -0.36687145]
སེམ	[0.18070808, -2.306222, 1.7001055, ……, 6.104775, 5.6365705, -5.3302755]
གཞུང	[4.0403185, 2.3763437, -6.7671647, ……, -0.02367509, -1.1939923, 1.8815073]
གིས	[-10.651783, 2.5300238, -2.4710336, ……, -3.2074265, 3.059835, -2.2537758]

3. Calculation of Similarity between Tibetan Syllables

In this paper, the CBOW model in Word2Vec was selected to represent Tibetan syllables as vectors, and through model training, each Tibetan syllable was represented as a 300-dimensional vector. When training the word vector model, an important hyperparameter of the model needs to be set, that is, the size of the window. In terms of setting the training window, when the window size is small, it can determine whether the syllables have similar characteristics in forming words or phrases, and the syllables have certain replaceable characteristics. When the window size is large, more semantic relatedness between syllables can be embedded.

The window sizes used in this article are 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, and 25. The models trained by Word2Vec are represented by $model_3$, $model_5$, $model_7$, $model_9$, $model_{11}$,$model_{13}$,$model_{15}$,$model_{17}$,$model_{19}$,$model_{21}$,$model_{23}$ and $model_{25}$, respectively. For example, the training window size of $model_3$ is 3, and during training, the model only focuses on the preceding and following one syllable of the target syllable to learn its relationship with the target syllable. The training window size of $model_{15}$ is 15, and during training, the model needs to focus on the preceding and following seven syllables of the target syllable to learn its relationship with the target syllable. The training window size of $model_{25}$ is 25, and during training, the model needs to focus on the preceding and following twelve syllables of the target syllable to learn its relationship with the target syllable.

When calculating the final similarity (represented by $\mathrm{Sim}_{\mathrm{output}}$) between two syllables, the following three steps are mainly used:

Step 1: Firstly, the similarity between syllables is calculated using the 12 different models mentioned above. The Word2Vec model's similarity function is mainly used to calculate the similarity between two syllables, which calculates the cosine similarity between the two vectors in the space. The similarity value ranges from -1 to 1. "-1" indicates that the two syllables are farthest apart, indicating that they have no similarity, while "1" indicates that the two syllables are closest, and when the two syllables are the same, their similarity value is 1. The expression for similarity calculation is: $\mathrm{S}_x = model_x.\mathrm{similarity}(\mathrm{syll}_a, \mathrm{syll}_b)$, where x represents the window size, syll_a and syll_b respectively represent the two syllables that need to be tested for similarity. When calculating the similarity between syllables, this article stipulates that syll_a is the problem syllable, which is the syllable that needs to be corrected or normalized, while syll_b is the correct syllable, which is a commonly used Tibetan syllable.

Based on the above expression, S_3 refers to the similarity value between syllables calculated using the $model_3$, which can reflect the semantic relatedness between two syllables in word formation. S_{15} refers to the similarity value between syllables calculated using the $model_{15}$. S_{25} refers to the similarity value between syllables calculated using the $model_{25}$.

Step 2: The $\mathrm{F\_Average}()$ function is used to calculate the average similarity at different levels. The main function of this function is to calculate the average similarity. If the number of input similarity values is 3, the average of the three values is directly calculated. If the number of input similarity values is 5, the method used is to first sort the group of similarity values, remove the maximum and minimum values, and then calculate the average of the remaining similarity values. In this step, a total of three groups of average similarity values are calculated, and their expressions are: $Sim_1 = \mathrm{F\_Average}(\mathrm{S}_{17}, \mathrm{S}_{19}, \mathrm{S}_{21}, \mathrm{S}_{23}, \mathrm{S}_{25})$; $Sim_2 = \mathrm{F\_Average}(\mathrm{S}_7, \mathrm{S}_9, \mathrm{S}_{11}, \mathrm{S}_{13}, \mathrm{S}_{15})$; $Sim_3 = \mathrm{F\_Average}(\mathrm{S}_3, \mathrm{S}_5, \mathrm{S}_7)$. Among them, the purpose of Sim_1 is to consider the influence of syllables that are far away from the center syllable, Sim_2 is to consider the influence of syllables with a moderate distance from the center syllable, and Sim_3 mainly considers the influence of syllables that are relatively close to the center syllable.

Step 3: The F_{opti} function is used in this article to analyze the three groups of average similarity values and the similarity value S_3, which were obtained in the second step, to calculate the final similarity value between syllables. The specific processing method of the F_{opti} function is as follows: First, calculate the average of the three groups of average similarity values, represented by $Sim_{average}$. If $Sim_{average}$ is greater than 0.2, it indicates that the two syllables have relatively high similarity. At this time, the function will output the three groups of average similarity values and the maximum value in S_3, because the maximum value can better reflect the degree of similarity between the two syllables. If the value of $Sim_{average}$ is in the range of [0-0.2], it indicates that the two syllables have moderate similarity. At this time, the function will output $Sim_{average}$. If $Sim_{average}$ is less than 0, it indicates that the two syllables have negative correlation. At this time, the function will output the minimum value in the three groups of average similarity values and S_3, because this can

better reflect the differences between the two syllables. This step is expressed as $\mathrm{Sim}_{output} = \mathrm{F}_{opti}(\mathrm{S}_3, Sim_1, Sim_2, Sim_3)$.

The similarity values mentioned in this article are all calculated using the expression Sim_{output}. The threshold value for syllable similarity is set to 0.1 in this article, which is the average value obtained by calculating each type of problem syllable. When calculating the similarity value between syllables, the similarity value of most syllables falls within the range of [-0.1, 0.1]. Therefore, if the similarity value between two syllables falls exactly within this range, it is difficult to determine whether it can be normalized.

4. Results of Syllable Similarity Experiment

This experiment mainly demonstrates the results of the Tibetan syllable similarity calculation experiment through three specific examples of syllable similarity.

4.1. Statistics of Syllables Compared for Similarity with Syllable "ཨ"

Table 3 provides information on the syllables related to the syllable "ཨ", including the corresponding definitions and similarity values between syllables. The syllables numbered 1-9 in the Table 3 are the syllables that are highly similar to the syllable "ཨ". It can be seen that these syllables mostly have strong semantic similarity with "ཨ". For example, the similarity between the syllable "ཨ" (father) and the syllable "ཡུམ" (mother) is 0.32648. The syllables numbered 10-12 in the Table 3 are the syllables that have low similarity with "ཨ", and their semantic similarity is weak.

Table 3. Information on Syllables Related to the Syllable "ཨ"

Number	Compared Syllable	Definition	Similarity	Number	Compared Syllable	Definition	Similarity
1	ཡུམ	Mother	0.32934	7	རྗེ	Elder brother	0.20035
2	ཡབ	Father	0.25695	8	མེས	Ancestor, grandfather	0.19862
3	ཁོ	Husband	0.23723	9	གདུང	Descendant	0.19084
4	གཅེན	Elder brother Elder sister	0.22869	10	ནགས	Forest	-0.05124
5	གཉུང	Younger brother Younger sister	0.22684	11	གནམ	Sky	-0.06157
6	སྤུན	Brother(s)	0.21692	12	གློག	Electricity	-0.06729

4.2. The Similarity between Syllables that can Form Words with the "སྙི"

The experiment uses the syllable "སྙི" (heart, wisdom) as the target syllable. Through similarity calculation, the related syllables that can form words with the target syllable are studied. "སྙི" is combined into 7 entries of words, and the entry information comes from Tibetan dictionaries.

In this paper, the first related syllable "བསམ" was taken as the benchmark and was compared with six other related syllables through similarity calculation. Specifically, the pre-trained Word2Vec models mentioned earlier ($model_3$,$model_5$,$model_7$,$model_9$, $model_{11}$, $model_{13}$, $model_{15}$, $model_{17}$, $model_{19}$, $model_{21}$,$model_{23}$,$model_{25}$)were used for similarity calculation.

Table 4. Examples of form words with the "ཟླ"

Related Syllables	Target Syllables	Combined words	Number of Texts	Definition of the words
བསམ	ཟླ	བསམ་ཟླ	13117	Thought, contemplation
འདོད	ཟླ	འདོད་ཟླ	1912	Willpower, proposition
མིག	ཟླ	མིག་ཟླ	1	Visual recognition, vision
རྨོངས	ཟླ	རྨོངས་ཟླ	2	Stupid thought
བྱིས	ཟླ	བྱིས་ཟླ	33	Childlike innocence
གོ	ཟླ	གོ་ཟླ	39	Wisdom, intelligence
དགྲ	ཟླ	དགྲ་ཟླ	0	Malicious, cruel-hearted

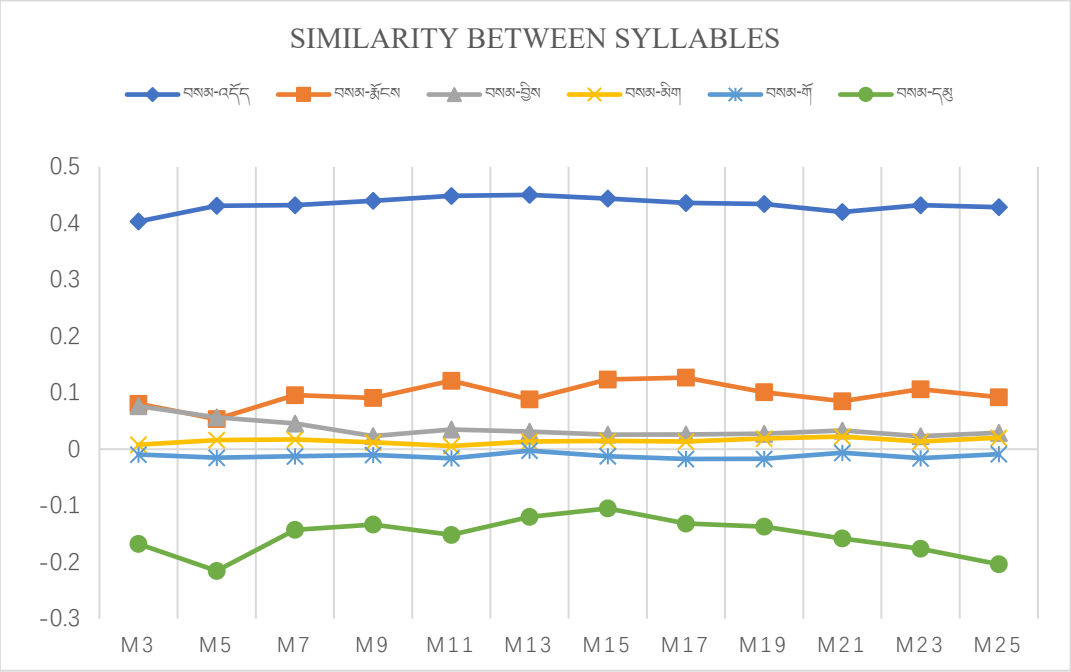

Figure 3. The similarity between syllables calculated by multiple window models

Figure 3 shows the performance of the similarity between syllables calculated by multiple window models. It can be concluded that not all combinations of syllables that have the ability to form collocations with the same syllable have high similarity. It is also necessary to consider the contextual semantic information of the combination of syllables in the text. For example, the syllables "བསམ" and "འདོད" have the highest similarity. From Figure 3, it can be seen that the similarity between these two syllables remains at around 0.4 under different window models. The high similarity between the syllables "བསམ" and "འདོད" indicates that they may have a high semantic correlation. For example, the two syllables may have similar collocation abilities, as shown in Table 5, where the term information is from the Tibetan dictionary. At the same time, it can be seen from Figure 3 that the similarity between the syllables "བསམ" and "དགྲ" is the lowest, and their average similarity is close to -0.15. This situation indicates that the semantic similarity between the two syllables is small, and there are significant differences between them in collocation ability and co-occurring syllables in the context. The similarity between the syllables "བསམ" and "དགྲ" increases slightly as the window of the model expands, indicating that the appearance of the same syllables in the context of these two syllables is affected by the increasing window of the model. However, these syllables are mostly function words that have little impact on the semantics of the text. The average similarity ranges of the other four

sets of syllables in Figure 3 are all between [-1, 1], indicating that there may be a certain semantic correlation between them.

Table 5. Collocation Phenomenon of Similarity between "བསམ" and "འདོད"

words	Number of Texts	Definition	words	Number of Texts	Definition
བསམ་དོན	193	mind; intention	འདོད་དོན	59	something one thinks about or seeks
བསམ་ཚུལ	2039	psychology; mood; emotions; thoughts	འདོད་ཚུལ	616	advocacy; proposition
བསམ་བྱོགས	3601	opinion; viewpoint	འདོད་བྱོགས	35	interest; ambition
ཕྱགས་བསམ	1687	ideal; aspiration	ཕྱགས་འདོད	2	ideal
མི་རབ་བསམ	2	ignorance; backwardness	མི་རབ་འདོད	0	obsession; infatuation

4.3. Judging the Similarity between Two Syllables through a Threshold Value

Table 6. Example Sentences in the Corpus Containing the Syllables "བསྒོ" and "བསྒོ"

Type	Sentences in Corpus
"བསྒོ" in Sentence	(1)མ་ཁས་པ་རྣམས་ཀྱིས་**བསྒོ**་བྱེང་གནང་བཞིན་པ།
	(2)ཀུང་གོའི་,,ཞིན་གསུམ་,,ཐོན་ལས་མཉམ་འདེས་འཛེལ་རྒྱས་གཏོང་བའི་མཐོ་རིམ་བྱེང་སྟེགས་སྐབས་གསུམ་པ་དང་།མགོ་འཚོང་མཉམ་ལས་ཁང་གིས་ཞིང་ཐེའི་དར་སྐེལ་འཛད་ཐུབ་ལ་ཞབས་འདེགས་ཞུ་བའི་ཚེད་དོན་**བསྒོ**་བྱེང་ཚོགས་འདུ་ཆེ་ལིང་ནས་བསྡུས།
	(3)དངོས་གཞིའི་སྐོབ་མ་སོགས་གང་ཡང་ཞུགས་ཚོག་པའི་རིགས་གཞུང་**བསྒོ**་བྱེང་ཞིན་པ་ལ་ངོས་འཛིན་དགོས་པའི་སྐོར།
	(4)ཟླ9པའི་ཚེས15ཉིན་གྱི་ཕྱི་དྲོ་གསལ་སྐྱོང་ལ་ཞུགས་པའི་སྐོབ་གཉེར་པ་ཁག་གསུམ་ལ་བགོས་ནས།,,ལ་ཁུལ་གསུམ་གྱི,,རིག་གནས་བྱ་བར་རྗེ་རྣར་སྐུལ་འདེད་ཕྱེད་དགོས་པའི་སྐོགས་ལ་**བསྒོ**་བྱེང་བྱས།
"བསྒོ" in Sentence	(5)རང་སྐྱོང་སྤྱོངས་མི་དམངས་འཕྲས་ཚོགས་རྒྱན་ལས་ཀུ་སྤྱན་གྱི་ཏང་ནུའི་རིགས་པའི་གཞུང་ལྷགས་སྐོབ་སྐྱོང་ཉེ་བའི་ཚོགས་རྩང(རྒྱ་ཚའི)གི་སྐོབ་སྐྱོང་**བསྒོ**་བྱེང་ཚོགས་འདུ་འཚོགས་པ།
	(6)འཕེལ་ཡོད་ཞེང་གཞིའི་ཐད་ནས་**བསྒོ**་བྱེང་གནང་བཞིན་པ།
	(7)སྤྱོངས་ཏང་ཨུད་ཏང་གི་སྐོབ་གྲས(རང་སྐྱོང་སྤྱོངས་སྡེད་འཛིན་སྐོབ་སྤྱིང)སྤྱོངས་ཡོངས་ཀྱི་ས་ཁུལ་དང་ཕིང་རིམ་པའི་འགོ་ཁྲིད་ལས་ཁུངས་པའི་ཚེད་དོན་**བསྒོ**་བྱེང་འཛིན་གྲྭ་དང་གཞོན་ནར་ལས་ཁུངས་པ་གསོ་སྐྱོང་འཛིན་གྲྭའི་འཛུགས་མིའི་བཞུགས་མིའི་བཀའ་ཤོལ་ཚོགས་འདུ་འཚོགས་པ།
	(8)རིགས་གཅིག་མཐས་ཅན་ཀྱི་**བསྒོ**་བྱེང་དང་། ལག་རྩལ་གསོ་སྐྱོང་། དེ་བཞིན་ས་གནས་ས་ཐོག་ཀི་མཐུས་ཁྲིད་སོགས་ཀྱི་ཁེད་སྐོ་སྤེལ་རྒྱུ་རེད།
	(9)བོད་ཞི་བས་བཅིངས་འགོལ་བཏང་ནས་ལོ་ངོ་70འགོར་བར་ཉེན་འཚེལ་ཞུ་བའི་རིག་གཞུང་**བསྒོ**་བྱེང་ཚོགས་འདུ་འཚོགས་པ།

When calculating the similarity between two syllables, if their similarity reaches a certain threshold value, the same syllables may appear in the context around these two syllables. In the experiment, "བསྒོ" and "བསྒོ" were selected as two example syllables, and Table 6 shows the syllables that appear in the context of both of these syllables, such as "ཞིང", "མཁས", "ཚེད", "དོན", "རིག", "གཞུང", "སྐྱོང", "ཚོགས", "འདུ", "གནང", "བཞིན", "འཚོགས", "ཞུགས", and so on. Meanwhile, through the similarity calculation, it was found that the similarity between the syllables "བསྒོ" and "བསྒོ" was very high, reaching 0.68337. These two syllables have similar appearances, with the only difference being the different subscripts in the syllables. All other Tibetan characters in their constructions are exactly the same. The syllable "བསྒོ" appeared 8 times in the 8 texts in the corpus, while the syllable "བསྒོ" appeared 4148 times in the 2422 texts in the corpus. In this article, "བསྒོ" is considered a problematic syllable, while "བསྒོ" is a commonly used syllable. Through similarity calculation, it can be concluded that the context language environment of a problematic syllable and the correct syllable have similarity. One of the research works in this article is to identify problematic syllables and to convert them to the correct syllables

through syllable normalization (the corresponding English translations of the Tibetan characters in the table are provided for the convenience of readers).

5. Conclusion

The main work of this article is focused on calculating the similarity between Tibetan syllables. When using the CBOW model to train Tibetan syllable vectors, different sizes of training windows are set to obtain semantic relatedness information at different levels of syllables, and multiple sets of word vector representation models are generated. Then, based on the similarity results calculated by multiple models, the average similarity between syllables is determined. The proposed method for calculating the similarity between Tibetan syllables can be applied to the syllable correction work in Tibetan corpus construction. When extracting syllables from large-scale Tibetan text corpora, there are often many irregular problems caused by input errors, syllable abbreviations, inconsistent character encoding, etc., such as missing syllable nodes, improper use of characters within syllables, missing characters within syllables, extra characters within syllables, splitting and spelling of independent encoded characters, and syllable abbreviation phenomena. We can statistically analyze the syllables in Tibetan corpora, classify and organize problematic syllables, and then use the proposed method for calculating the similarity between Tibetan syllables to develop corresponding syllable correction plans for different types of problematic syllables, thereby completing the construction of a universal Tibetan syllable library. By using the universal Tibetan syllable library, we can evaluate the quality of Tibetan texts. Through text quality assessment, higher-quality text corpora can be provided for related tasks in intelligent processing of Tibetan texts.

Acknowledgments

This research received the support of the Science and Technology Planning Project of Gansu Province (NO. 23JRRA1736) and the Central University Fundamental Research Funds Special Fund Project of Northwest Minzu University (Grant NO. 3192023030). Additionally, it was also supported by the 2023 Open Laboratory Projects of Northwest Minzu University (Grant NO. SYSKF-2023060).

References

[1] Tibetan language, Baidu Baike, https://baike.baidu.com/item/%E8%97%8F% E8%AF%AD/.

[2] Cai Zhijie, CaiRang Zhuoma. Design of a Tibetan Word Attribute Analysis System Based on Corpus. Computer Engineering, 2011, 37(22): 270-272.

[3] Bai Xuejun, Gao Xiaolei, Gao Lei, Wang Yongsheng. Eye Movement Study on Perception Breadth of Tibetan Reading. Acta Psychologica Sinica, 2017, 49(05): 569-576.

[4] Zhu Feng. Three Revisions of Tibetan Standardization. Language and Writing News, 2014, 493.

[5] Cai Rang Zhuoma, Cai Zhijie. Modern Tibetan Word Building Decomposition Method. Journal of Qinghai University, 2010, 28(4): 83-86.

[6] Jiang Di, Dong Yinghong. Statistical Research on Tibetan Information Processing Attributes. Journal of Chinese Information Processing, 1995, 9(2): 37-44.

[7] Gesang Jumian, Gesang Yangjing. Practical Tibetan Grammar. Chengdu: Sichuan National Publishing House, 1987.

[8] Kong Jiangping, Yu Hongzhi, Li Yonghong, Dawapengcuo, Hua Kan. Tibetan Dialect Survey Form. Beijing: Commercial Press, 2011.

[9] Wang Fuzhao, Zhou Yan. Research on Error Detection Method of Tibetan Syllables. Computer Era, 2020(01): 5-9.

[10] Zhu Jie, Li Tianrui, Liu Shengjiu. TSRM Tibetan Spelling Check Algorithm. Journal of Chinese Information Processing, 2014, 28(3): 92-98.

[11] Duola, Zaxijia. Standard Syllable Frequency of Tibetan. Beijing: China Social Sciences Press, 2015.

[12] Liu Huidan, Hong Jinling, Nuo Minghua, et al. Statistical Analysis of Tibetan Syllable Spelling Errors Based on Large-scale Network Corpus. Journal of Chinese Information Processing, 2017, 31(2): 61-70.

[13] Zheng X, Chen H, Xu T. Deep Learning for Chinese Word Segmentation and POS Tagging[C]. Conference on Empirical Methods in Natural Language Processing. 2013:647-657.

[14] Quoc V. Le, Tomas Mikolov. Distributed Representations of Sentences and Documents. arXiv:1045.4053,2014.

[15] Mikolov T, Chen K, Corrado G and Dean J. Efficient Estimation of Word Representations in Vector Space, arXiv preprint arXiv:1301.3781, 2013.

[16] Devlin J, Chang M W, Lee K, et al. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding[C]. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics, 2019: 4171-4186.

[17] Cai Zhijie, Sun Maosong, Cai Rang Zhuoma. Construction of Evaluation Set for Tibetan Word Vector Similarity and Correlation. Journal of Chinese Information Processing, 2019, 33(07): 81-87.