User Preference Model-Based Sentiment-Semantic Corresponding Method For Korean Text Translation

Lu Cong*

School of Northeast Asia, Liaoning University of International Business and Economics, Dalian, 116052, China

*Corresponding author. E-mail: lucong054@gmail.com

Received: July. 28, 2025; Accepted: May. 25, 2026

In order to solve the problems of poor matching accuracy and long time-consuming in traditional Korean text translation emotional semantic matching methods, this paper proposes research on a Korean text translation emotional semantic matching method considering a user preference model. Build a user preference model, input that obtained emotional space position perception vector into the user preference model, correcting the translation semantic result according to the user's emotional preference, determining the freshness of keyword features, calculating the freshness weight, generating user preferences, and calculating the similarity of interest vectors among users, on this basis, obtaining a Korean text data set, and using Word2Vec model to map individual words in natural language corpus to vectors to extract Korean text translation features; By inputting word vectors into DPCNN, depth features are extracted to classify text emotions. The semantic matching model of translated text is constructed by sentence representation, and the emotional space position perception vector is introduced to deeply match the emotional semantics of Korean translated text, which makes the semantic matching result more accurate. The experimental results show that the semantic matching accuracy of Korean translation text under this method is 99.6%, and the matching time is 3.6s, which can effectively improve the accuracy of emotional semantic matching and further improve the translation effect of Korean text.

Keywords: User preference model; Freshness; Korean text; Emotional semantics; Semantic matching

http://dx.doi.org/10.6180/jase.202610_33.028

1. introduction

Cross-language communication has become an indispensable part of daily life and work. As a bridge between Korea and the rest of the world, Korean text translation is self-evident. However, simple text translation has been difficult to meet the growing demand [1]. Users not only require the accuracy of translation, but also expect the translation results to accurately capture the emotional color of the original text, taking into account the timeliness or "freshness" of the content [2]. Therefore, the emotional and semantic matching technology of Korean text translation, which combines a user preference model and freshness factor, came into being and became an important research direction in the field of natural language processing [3]. User preference model refers to a model that can reflect the personalized characteristics of users by collecting, analyzing and understanding information such as users' behaviors, interests and needs [4]. In Korean text translation, the application of a user preference model can significantly improve the personalization of translation results. Different users may have different preferences for the translation of the same sentence: some users prefer literal translation to preserve the original style, while others hope that the translation will be more fluent and natural, and even integrate into the expression habits of the target language [5]. By constructing the user preference model, the translation system can intelligently adjust the translation strategy according to the

user's historical behavior and current needs, and provide translation results that are more in line with the user's expectations. The consideration of freshness means that in the rapidly changing information age, the freshness of text content is directly related to its value and attractiveness. For Korean text translation, maintaining the freshness of translation content means timely capturing and accurately conveying new words, new expressions and hot topics in the original text. This requires that the translation system should not only have strong language processing ability, but also have keen insight into current events, culture and social dynamics. By combining timestamp, keyword popularity, social media trends and other data, the translation system can evaluate the freshness of the original content and give appropriate treatment in the translation process to ensure that the translation can not only accurately convey the original information, but also reflect its timeliness.

Couairon and colleagues [6] introduced a novel approach for sentiment-semantic matching in Korean text translation, utilizing a deep generative model. The initial step involved creating a dataset of Korean texts paired with sentiment labels. Subsequently, various preprocessing tasks were conducted, including data cleaning, segmentation, and annotation. To enhance the model's capability in processing emotional semantics, an emotion encoder was incorporated along with an emotion classifier to learn the emotional-semantic representation within the text. Utilizing an emotion loss function enforced constraints on the model, ensuring precise sentiment-semantic matching throughout the translation process. Despite the method's success in enhancing sentiment-semantic matching precision, the challenge of acquiring highquality Korean text datasets with sentiment labels persisted, potentially impacting the efficacy of the model's training. Batbaatar et al. [7] introduced a new neural network structure known as the Semantic-Emotion Neural Network (SENN), which integrated semantic/syntactic and emotional data by incorporating pre-trained word embeddings. The SENN architecture merged convolutional neural networks (CNNs) capabilities with bidirectional Long-Short Term Memory (BiLSTM) networks. The BiLSTM module captured contextual details and highlighted semantic connections, whereas the CNN module extracted emotional characteristics and concentrated on the emotional associations among words. This dual methodology empowered SENN to proficiently analyze and interpret intricate linguistic and emotional subtleties within the text. Pieroni et al. [8] developed strategies for training emotional-semantic representation to help the model grasp emotional content in text. They supported the model in comprehending emotional semantics through the utilization of sentiment dictionaries and sentiment classifiers. Furthermore, they established an emotional-semantic matching system to ensure that the translated text maintains the emotional subtleties of the source language, attaining emotional-semantic alignment in Korean text translation by integrating an emotional loss function. Nevertheless, because emotional expression can differ across languages, the translated text may display inconsistent emotional semantics, resulting in diminished emotional precision. Yu et al. [9] created the innovative smart mirror LUX to enhance mental well-being by integrating Artificial Intelligence (AI) technology. The researchers addressed the absence of Korean emotion databases and the challenges posed by language disparities between Korean and English. LUX employs sophisticated deep learning (DL) algorithms to interpret emotions such as anger, sadness, neutrality, and happiness, providing users with various responses, including inspirational quotes, music selections, and empathetic interactions. The smart mirror also offers a wake-up alarm, weather forecasts, calendar reminders, real-time news updates, and a digital clock. Zhang et al. [10] suggested an emotional-semantic matching method for Korean text translation focusing on structural and semantic features. They collected a Korean text dataset and extracted the structural and semantic characteristics of the text. Semantic features in the learning text were displayed utilizing the emotional word embedding method. These emotional-semantic features of Korean text translation were then extracted and analyzed to construct an emotional-semantic matching model, achieving effective emotional-semantic matching in text translation. However, this approach is constrained by computing resource limitations. Cheng and Kuang [11] introduced a technique for emotion-semantic alignment in Korean text translation through association matching. They gathered and analyzed a dataset of Korean text and formulated a Seq2Seq model to tackle the issue of sequence-to-sequence conversion. An attention mechanism was implemented to enhance the model's understanding and alignment of emotional semantics. Uymaz et al. [12] investigated how texts can reflect emotions and the difficulties in detecting emotions through conventional word representation patterns such as Word2Vec and GloVe.

They observed that including emotional data in word vectors enhanced the precision of emotion detection. This review analyzed current research, outlining the methodologies, emotion patterns, datasets, and outcomes. Ahmad et al. [13] introduced a method that integrated a Multichannel CNN (MC-CNN) with a multichannel bidirectional gated recurrent unit (MC-BiGRU), with equal input parameters

for both. Furthermore, sharing hidden layer information between these concurrently integrated frameworks enhanced their collective capabilities, capitalizing on the advantages of both methodologies. Liu et al. [14] proposed a sentiment analysis approach drawing on DL to produce probabilistic linguistic term sets (PLTSs) from online product reviews and rank them. They utilized a natural language processing technique to extract product attributes and texts and subsequently employed sophisticated DL patterns for sentiment categorization. They implemented a matching mechanism to synchronize sentiment trends with suitable linguistic terms for each rating category to guarantee precision. Park et al. [15] introduced a cross-modal Semantic Alignments Module (SAM) to improve inter-modal connections and create apparent alignments. Initially, visual and textual features were obtained from pairs of images and text. Subsequently, a bipartite graph was created, where image regions and words from the sentence served as nodes, and their connections were displayed as edges. The model could calculate attention weights through SAM using information from these graph edges. Jiang and Cai [16] sought a succinct yet through examination of the text-matching domain. They summarized the key concepts, challenges, and resolutions for using statistical methodologies and neural networks for text-matching approaches. Additionally, they explored matching techniques leveraging extensive language patterns, scrutinizing the setups and utilities of Application Programming Interfaces (APIs). They also deliberated on pertinent datasets and assessment methodologies to provide a comprehensive insight into the present status and progressions in text matching.

Wang et al. [17] suggested in 2025 a unique dataset that consists of issues that are related to doors found when the houses were delivered. There are eight types of issues in this dataset. This is an interesting dataset that uses real world data that was collected from three different apartment buildings and makes use of classifications by experts. Wang [18] tested five text classification models that relied on transformers including BERT, RoBERTa, ALBERT, DistilBERT, and XLNet in order to detect defects in fire doors in residential buildings. RoBERTa algorithm demonstrated the best results when optimized, with an average F1-score reaching 85.44%. Wang [19] created and evaluated four different text classification models using Graph Neural Networks (GNN) that could be used automatically to recognize defects in the fire-doors installed in residential buildings. Of the models mentioned above, the model BERT-GCN demonstrated the best results, reaching an F1 score of 85.46%, outperforming 2,430 different models.

In view of the above problems, this paper puts forward the emotional semantic matching of Korean text translation, considering the user preference model and freshness. Emotional semantic matching is an advanced task in natural language processing, which requires that the translation system can not only understand the literal meaning of the text, but also accurately grasp the emotional color in the text, and maintain this emotional consistency in the translation process. Emotional semantic matching is particularly important in Korean text translation. Because Korean is a language rich in emotional expression, its texts often contain rich emotional information. If the translation system cannot accurately capture and convey this emotional information, then the translation will lose its original charm and appeal. Therefore, by introducing sentiment analysis technology and a semantic matching algorithm, the translation system can deeply understand the emotional tendency of the original text and express it in the translation, thus improving the overall quality of translation.

In this paper, a user preference model is constructed, and the obtained emotional space position perception vector is input into the user preference model. According to the user's emotional preference, the translation semantic results are corrected, the freshness of keyword features is determined, the freshness weight is calculated, the user preference is generated, and the similarity of interest vectors among users is calculated. On this basis, the Korean text data set is obtained, and the single words in the natural language corpus are mapped to vectors by the Word2Vec model to extract Korean text translation features. By inputting word vectors into DPCNN, depth features are extracted to classify text emotions. The semantic matching model of translated text is constructed by sentence representation, and the emotional space position perception vector is introduced to deeply match the emotional semantics of Korean translated text, which makes the semantic matching result more accurate. The innovation of this paper lies in: for the first time, the user preference model is combined with the freshness mechanism and introduced into the emotional semantic matching task of Korean text translation. Traditional methods mostly focus on a single aspect of semantics or emotion, and fail to fully consider the influence of users' personalized preferences and content timeliness on translation results. The method proposed in this paper not only realizes the personalized correction of translation results by constructing a dynamic user preference model, but also introduces a freshness weight mechanism to simulate the decay process of user interest over time, thus more accurately reflecting the current emotional and semantic needs of users.

2. methods

Input the emotional space position perception vector obtained in the previous section into the user preference model, and correct the translation semantic result according to the user's emotional preference, so that the semantic matching result is more accurate. The core innovation of this section is to combine the emotional space position perception vector with the user preference model to realize the dynamic correction of translation semantic results. By introducing the layered modeling of long-term preference and short-term preference, and the weight attenuation mechanism based on freshness, this method can capture the changing trend of users' interests more finely, and then improve the accuracy and timeliness of emotional semantic matching.

2.1. User preference model framework

In order to dynamically and accurately reflect user interests and preferences, user interests are divided into long-term and short-term preferences. Long-term preference refers to an individual's preference for a certain thing or field over a relatively long period of time. Longterm preferences are relatively stable and closely related to personal attributes such as knowledge accumulation, life experience, and personality. Short-term preference refers to an individual's interest in certain things over a relatively short period of time due to environmental factors and external stimuli. The framework diagram of the user preference model is shown in Fig. 1:

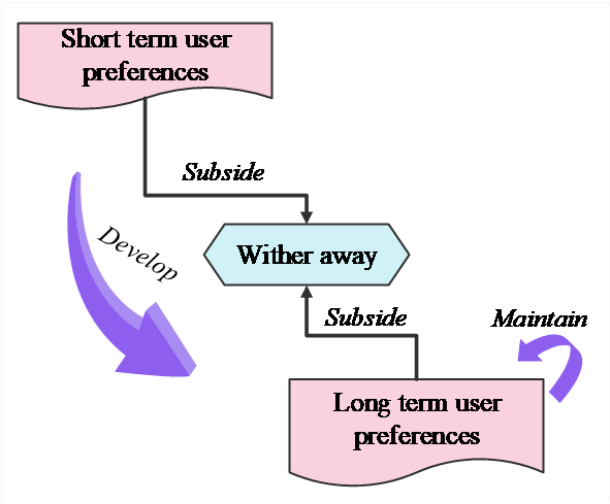

Fig. 1. User preference model framework diagram

User preferences and interests can develop into long-term preferences, or short-term preferences gradually fade and eventually lose interest. Although long-term preferences can be maintained consistently, they may fade progressively into disinterest.

The definition of user long-term preferences is: Due to the stability of long-term preferences and the potential involvement of many types of information, this information can be classified, with each category corresponding to a field of interest for the user. Therefore, long-term preference can be expressed as:

$$L = \{(l_1, l_1, q_1), (l_2, l_2, q_2), \ldots, (l_n, l_n, q_n)\} \tag{1}$$

Here, l_i signifies a field that the user is interested in, σ_i indicates the degree to which the user is interested in the field, q_i implies the number of Korean documents in the field that the user is interested in. σ_i is defined as:

Submission Template to Journal of Applied Science and Engineering

$$\sigma_i = q_i / \sum_{i=1}^{n} q_i \tag{2}$$

The long-term preferences of users encompass multiple interest fields, where each domain l_i maintains a feature keyword library $(k_1, k_2, \ldots, k_n)$ to describe the Korean document domain.

The definition of short-term preferences is: Due to the continuous advancement and changes of short-term interests, except for a small amount of conversion into long-term interests, more short-term interests always appear quickly and then disappear soon, so it is not suitable to classify short-term interests and treat all short-term preferences as a whole. Define user short-term preferences:

Short-term preference S is displayed by a set of keyword tuples $S = (s_1, s_2, \ldots, s_m)$, keyword tuples $s = (k, v)$, where k signifies the keyword and v represent the weight corresponding to that keyword. Users will have both long-term and short-term preferences. User preferences are defined as $P = (S, L)$, where S represents short-term preferences and L represents long-term preferences.

2.2. Determination of the freshness of keyword features

The notion of freshness is introduced, and a calculation formula for simulating user interest decay over time is suggested. Freshness represents the gradual decrease in the attractiveness of items or preferences to users over time, reflecting how specific short-term preferences lose their appeal gradually.

Freshness is defined as a function that decreases over time. As time progresses, the value of freshness gradually diminishes, eventually reaching 0 , indicating that the user no longer finds the item or preference appealing. The specific definition is:

$$r_t = \begin{cases} 0, & \text{if } 0 \le r_{t_0} \le \varepsilon \\ \left(\frac{r_{t_0}-\varepsilon}{r_{t_0}}\right)^{t-t_0}, & \text{if } r_{t_0} > \varepsilon \end{cases} \quad (3)$$

If t_0 is a specific time in the past and t is the current time, then r_{t_0} displays the freshness of the tuple at time t_0, and r_t implies the freshness at time t. The freshness at the current time is calculated using r_{t_0} and Eq. (27). Here ε is a small constant used to determine when freshness is sufficiently low that the new freshness value is considered 0 . Initially, the freshness value of a tuple at its creation time is set to 1 .

Obtain all log documents for this user. If the log documents have been newly added or recently modified by the user, re-segment and extract feature words from the papers. Next, calculate the word frequency statistics of the keywords in each document. Finally, generate a tuple set by calculating the freshness value of each document. If the log document has already been processed, update its freshness value to update its tuple set. The design diagram of this process is displayed in Fig. 2

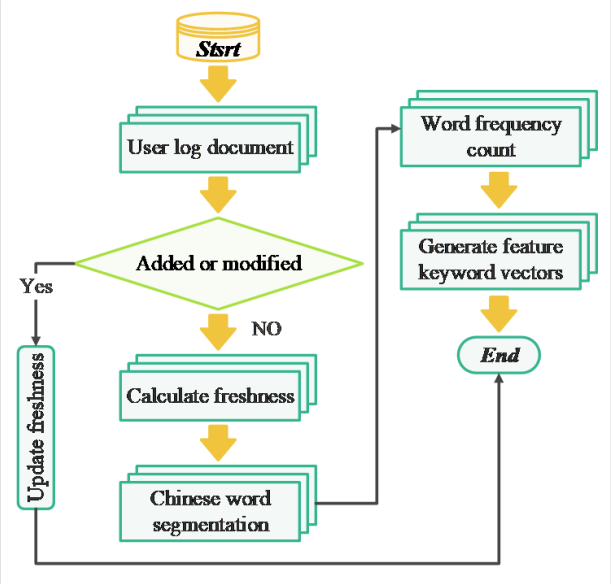

Fig. 2. Flowchart of feature keyword vector extraction for user documents

2.3. Freshness Weight Calculation

Any keyword's weight should decrease over time due to the influence of freshness. Therefore, during the update process, the real-time weight of the keyword is calculated as the product of its initial weight and freshness. For a particular keyword, this relationship can be expressed as:

$$(k_p, v_1, t_1, d_1, r_1), (k_p, v_2, t_2, d_2, r_2), \ldots, (k_p, v_n, t_n, d_n, r_n) : d_1, \ldots, d_n$$

displays a document in which a particular keyword has appeared. Since user preferences are expressed using keywords, it is crucial to understand the importance of each keyword to the user's interests. Therefore, the total weight of keyword k is the weighted sum of the corresponding weights and freshness of each document: $v_p^t = v_1 \cdot r_1' + v_2 \cdot r_2' + \ldots + v_n \cdot r_n', r_1', r_2', \ldots, r_n'$ signifies the current freshness of the corresponding keywords in each document. This cumulative weight is defined as the final weight of the keyword at time t.

After calculating the weight of each keyword at time t, the user's short-term interest vector can be generated. The steps are outlined in Algorithm 1:

Algorithm 1: User Preference Generation Algorithm

Input: Set of feature keyword vectors

Output: User preference vector $S = ((k_1, v_1), (k_2, v_2), \ldots, (k_m, v_m))$.

While the User keyword tuple set

Start

Find all tuples with the exact keywords in the keyword tuple set

Calculate keyword weight $v_p^t = v_1 \cdot r_1' + v_2 \cdot r_2' + \ldots + v_n \cdot r_n'$.

End

After calculating the weights of all keywords, the next step is to establish or update user preferences by selecting the keywords with the highest weights and their corresponding values. This results in $S = ((k_1, v_{\max_1}), (k_2, v_{\max_2}), \ldots, (k_m, v_{\max_m}))$. The value of m was determined through extensive experimental testing. In this paper, $m \le 3$ was chosen, meaning that the top three keywords that best reflect user interests are selected to display as user preferences. Fig. 3 illustrates the process design diagram for the above steps.

Based on the above, it is clear that the final weight of keywords undergoes a continuous accumulation process. With the addition of new information, these weights will continue to increase. Conversely, in the absence of new information, the current weights of keywords gradually decrease due to diminishing freshness. This reflects a decrease in user interest in those particular keywords. Over time, as weights are recalculated for each keyword, those with higher weights may indicate long-term interests, while lower weights may suggest a decline in user short-term preferences.

2.4. Emotional-Semantic Matching Correction in Korean Text Translation

User preferences are generated based on the user preference model $p = (s, L)$:

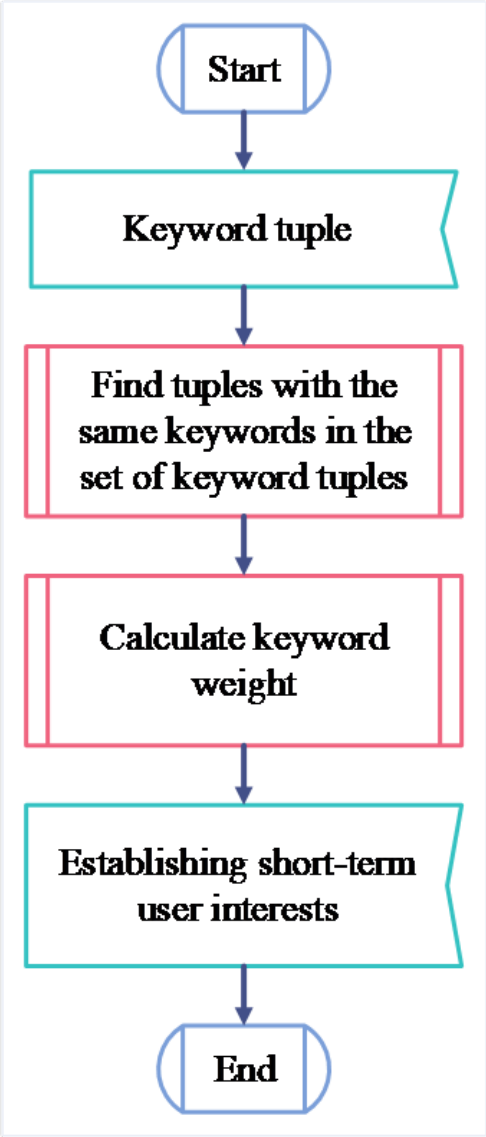

Fig. 3. Generate preference flowchart

$$s = \{(k_1, v_1), (k_2, v_2), \ldots, (k_m, v_m)\},$$
$$L = \{(l_1, \sigma_1, q_1), (l_2, \sigma_2, q_2), \ldots, (l_n, \sigma_n, q_n)\}.$$

To calculate the similarity of interest vectors between users, for a user's preferences p, Calculate the similarity with another user p_i [20] :

$$\text{sim}\left(\vec{P}, \vec{P}_l\right) = \frac{\vec{P} \cap \vec{P}_l}{|\vec{P}| * \left|\overrightarrow{P_l}\right|} \tag{4}$$

Using Eq.(4), identify the user set $R = \{p_{\max_1}, P_{\max_2}, \ldots, P_{\max_m}\}$ that is most similar to the user's preferences. Utilize the user preference model to provide feedback on the user's emotions within this set to adjust semantic matching outcomes, achieving semantic matching correction. The parameter m was determined through extensive experimentation to find a suitable threshold. In this system, $m \leq 3$ was selected.

3. emotional semantic matching in korean text translation

3.1. Feature extraction for Korean text translation

Firstly, the Korean text dataset is obtained using OPUS (http://opus.nlpl.eu/). OPUS is an open parallel corpus that contains translation data for multiple language pairs, including Korean. Organize the collected and processed data into a dataset format and convert the data from parallel corpora into a CSV file format. Utilizing the bag-of-words model, individual words in the natural language corpus are mapped to vectors to obtain Korean text word vectors, the features of Korean text translation. The bag-of-words model involves putting these words into a "bag" after text segmentation, treating them as independent entities, and constructing a dictionary that includes all the words in the text [21]. This article applies the Word2Vec bag-of-words model to map words with similar meanings into similar vectors, resulting in similar representations. This helps machines learn the definitions and contexts of different words.

Word2Vec maps individual words from natural language corpora to vectors [22] to obtain model parameters through training and use these parameters as word vectors, as displayed in Eq. (1).

$$f(w_t, w_{t-1}, \ldots w_{t-n+2}, w_{t-n+1}) = p\left(w_t, w_t^{t-1}\right) \tag{5}$$

The word vector model needs to satisfy the conditions displayed in Eqs. (6-7):

$$f(w_t, w_{t-1}, \ldots w_{t-n+2}, w_{t-n+1}) > 0 \tag{6}$$
$$\sum_{i=1}^{|V|} f(w_t, w_{t-1}, \ldots w_{t-n+2}, w_{t-n+1}) = 1 \tag{7}$$

The word vector model converts input words into vector representations, which can also be understood as a mapping relationship. This mapping is denoted by C, so $C(w_{t-1})$ signifies the word vector of w_{t-1}. The expression $p\left(w_i = i \mid w_i^{i-1}\right)$ displays the probability of the i th element in the vector. In the model, it is necessary to maximize the value of the logarithmic likelihood function [23], as displayed in Eqs. (8) and (9).

$$L = \frac{1}{T}\sum_{t}\log f\left(w_t, w_{t-1}, \ldots, w_{t-n+1}; \theta\right) + R(\theta) \quad (8)$$

$$f(x) = b + Wx + U\tanh(d + Hx) \quad (9)$$

Among them, $w_t, w_{t-1}, \ldots, w_{t-n+1}$ refers to the input of the model, and the word vectors $C\left(w_{t-1}\right)$ for the remaining words can be obtained through C. To control the range of output values, the output of the word vector model is input into the SoftMax function, as shown in Eq. (10).

$$p\left(w_t \mid w_{t-1}, \ldots, w_{t-n+1}\right) = \frac{e^{y_{w,i}}}{\sum_i e^{y_i}} \quad (10)$$

The input text corpus is mapped to the mapping layer through accumulation, and the model has no hidden layer. Therefore, after the text passes through the Continuous Bag of Words (CBOW) model, the output is the semantic information of the text sequence [24–27], which significantly enhances the computational speed. The CBOW model processes the obtained vector through an activation function to get the probability formula for the target word. The word with the highest probability is the predicted target word, as displayed in Eq. (11).

$$P(\omega \mid c) = \frac{\exp\left(e'(w)^T x\right)}{\sum_{w' \in V}\left(e'\left(\omega'\right)^T x\right)} \quad (11)$$

The Skip-Gram model first obtains a word vector from the vocabulary [28], and then displays the surrounding context word vector through the word vector of the target word. Its objective function is displayed in Eq. (12).

$$\max\left(\sum_{(\omega,c)\in D}\sum_{\omega_j \in c}\log P\left(\omega \mid \omega_j\right)\right) \quad (12)$$

This completes the feature derivation of Korean translation using Word2Vec.

3.2. Korean Text Sentiment Classification

Deep Pyramid Convolutional Neural Network (DPCNN) is a low-complexity deep convolutional neural network (DCNN) that can capture interrelated information from text and obtain richer and deeper emotional information in the text. Moreover, the computational complexity of each layer of this model is significantly reduced compared to traditional patterns.

DPCNN mainly uses text region embedding for word vector representation, which is unsuitable for handling significant data situations. Simultaneously, this word vector representation method cannot fully understand the emotional information in the text. Considering that the fusion of multiple model attributes can improve the classification performance and effectiveness of the model, this section obtains word vectors. Compared to using DPCNN text region embedding to obtain word vector representations, it can capture richer semantic information in the text [29]. After receiving word vectors through the Bidirectional Encoder Representations from Transformers (BERT) model, the word vectors are input into DPCNN for deep feature derivation. The BERT model is combined with DPCNN to complete the task of text sentiment classification, further enriching the emotional-semantic representation of the text. The structure of the text sentiment classification algorithm drawing on BERT-DPCNN is displayed in Fig. 4, which includes an input layer, a feature derivation layer, a pooling layer, a fully connected layer, and an output layer.

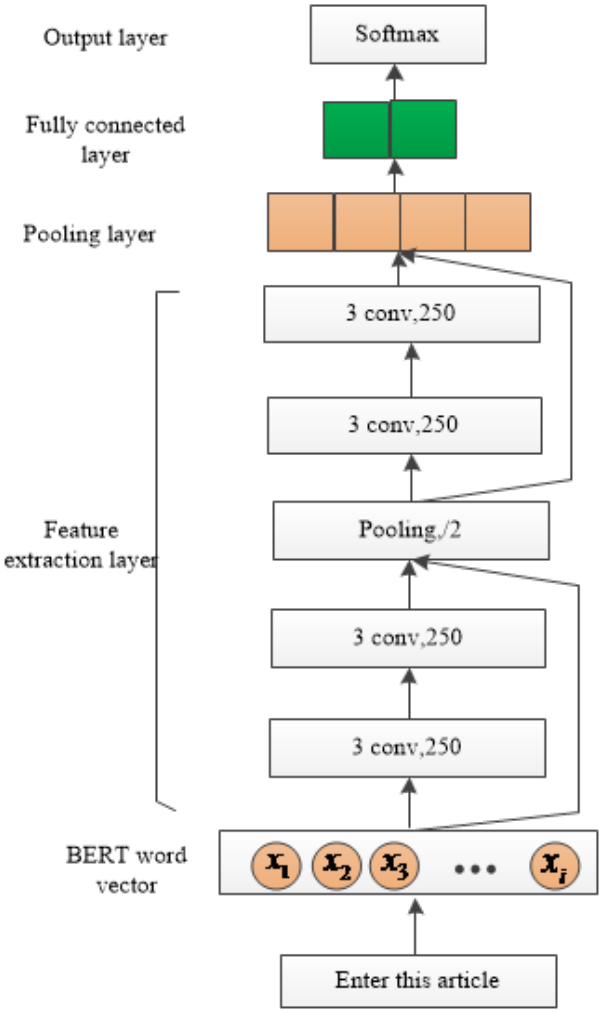

Fig. 4. Structure diagram of the BERT-DPCNN text sentiment classification algorithm

The first layer is the input layer, which inputs the sentiment dataset into the word vector model. This section uses the BERT word vector model. The BERT model has

already learned the emotional features contained in the text through the pre-training stage, so the classified text can obtain word vectors containing the emotional features of the text after passing through the BERT model [30].

The BERT-DPCNN model first converts each word in the input text into a word vector X_i, as displayed in Eq. (13), where X_{ni} implies the word vector of the n-th word in the i-th sentence. Then, the word vectors are concatenated to obtain a word vector matrix, as displayed in Eq. (14).

$$X_i = (X_{1i}, X_{2i}, \ldots, X_{ni}) \tag{13}$$

$$X_i = (X_{1i} \oplus X_{2i} \oplus \ldots \oplus X_{ni}) \tag{14}$$

CNNs can obtain local semantic information of the text to be classified through convolutional operations, but cannot obtain long-distance semantic details of the text to be classified. Therefore, DPCNN deepens the network to get a more profound semantic representation richness of the text to compensate for the disadvantage of CNNs not being able to obtain long-distance semantic information of the text to be classified, and to achieve better classification outcomes. Input the word vector matrix X output by the BERT model into the DPCNN model, and a DCNN for equal-length convolution is used to generate text features. Equal-length convolution means that the sequence length remains unchanged after the convolution operation. In this model, the size of convolution strides is indicated in former and subsequent convolutions by a parameter equal to 3 . This means the current word can obtain semantic information from the three adjacent words following a convolution of the same length. This way, the current word contains contextual semantic information, thereby obtaining deeper semantic information. The text features generated by the word vector matrix X are displayed in Eq. (15), where b denotes the deviation and f displays the ReLU nonlinear activation function [31]. The final obtained text features are displayed in Eq. (16).

$$C_i = f\left(WX_i + b\right) \tag{15}$$

$$C = C_1 \oplus C_2 \oplus \ldots \oplus C_n \tag{16}$$

The DPCNN model performs two equal-length convolution operations on the concatenated word vector matrix, then adds the word vector matrix to the result of the equal-length convolution, performs a $1/2$ pooling operation, and repeats the equal-length convolution operation. The maximum pooling operation is performed on the result of equal-length convolution, as displayed in Eq. (17). The maximum pooling operation can alleviate the problem of vanishing or exploding gradients caused by small weights [32–34]. Finally, the obtained text sentiment information is concatenated and linearly transformed through a fully connected layer, which is then passed through a SoftMax layer to output the predicted sentiment classification polarity. Positive and negative polarities represent the two classification outcomes, and the sum of polarities is 1, as displayed in Eqs. (18) and (19). Among them, i displays the polarity of sentiment classification, and $z_i = WX_i + b$.

$$P_{\max} \tag{17}$$

$$\sigma_i(z) = \frac{e^{z_i}}{\sum_{j=1}^{m} e^{z_j}} \tag{18}$$

$$\sum_{i=1}^{j} \sigma_i(z) = 1 \tag{19}$$

The loss function of the model adopts Cross Entropy loss function. The cross-entropy loss function can effectively reflect the deviation between the model prediction probability distribution and the real label. The smaller the cross entropy, the higher the accuracy of model classification, as shown in Equations (20) and (21).

$$\text{loss} = -a\frac{1}{n}\sum_{i=1}^{n} y_i(\ln a) + (1 - y_i)\ln(1 - a) \tag{20}$$

$$P(y = 1 \mid x) = a, a = \frac{1}{1 + e^{-x}} \tag{21}$$

In model optimization, we use the Adam optimizer. Adam optimizer combines the advantages of momentum and adaptive learning rate, and can usually achieve fast and stable convergence in practice. The loss function is minimized by the Adam optimizer, and the weight and bias parameters of the model are iteratively updated to obtain the best classification effect. Drawing on this, the classification outcomes of Korean translation texts are obtained, and based on them, the emotional-semantic matching of Korean translation texts is performed.

3.3. Deep Emotional-Semantic Matching of Korean-Translated Texts

After obtaining the feature derivation outcomes of the Korean translation article, analyze the emotional tendency of the Korean translation article, extract the emotional tendency features of the Korean translation article, and achieve deep emotional-semantic matching of the Korean translation text.

Construct a semantic matching model for translated text through sentence representation, use the Siamese architecture to encode the overall semantic information of sentences, construct a complete sentence representation, and

then achieve sentence matching. Specifically, the model first uses a representation layer to encode sentence vectors, then a matching layer to calculate the degree of correlation between two statement representation vectors, and finally a decision layer to infer the matching relationship between the two statements. The most crucial part is the representational construction of text statements. As much sentence information as possible should be integrated into the sentence vector in constructing text-statement representations. High-quality statement representation is of great help for subsequent matching analysis and can also affect the performance of the matching model.

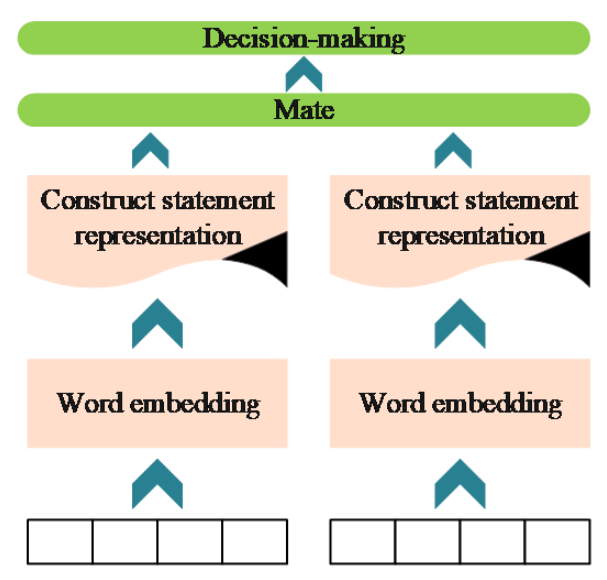

Fig. 5. Text statement matching drawing on statement representation

The advantage of a matching model based on sentence representation is that it can preprocess text data, and the model is better at understanding the overall semantic information of sentences. The disadvantage is that the semantic perception ability of local words is weak, and it is easy to overlook the corresponding relationships between essential words. Considering that most high-performance patterns in existing statement representation schemes draw on attention mechanisms to represent statement vectors, the model suggested in this paper draws on self-attention mechanisms to construct a representation of statements. However, the selfattention mechanism is imperfect, and there are still some problems in constructing vector representations of statements drawing on this approach. Therefore, this article introduces emotional spatial position perception vectors for deep emotional-semantic matching of Koreantranslated texts.

When constructing sentence representation, this paper introduces an emotional space position perception vector to achieve deeper matching. The core of this vector is to simulate the common influence of emotional differences and relative positions between words on semantic understanding when reading.

Emotional difference: refers to the degree of difference between any two words in a sentence in terms of emotional polarity. Calculate an emotional intensity score for each word through a pre-trained emotional dictionary (for example, positive words are +1 , negative words are -1 , and neutral words are 0). Meaning is the absolute value of the difference between their emotional intensity scores. The smaller the emotional difference, the more consistent the emotional tendency of the two words, and the higher the possibility that they express a certain point of view together in semantics.

Location information: refers to the relative distance between words in a sentence. Given a central word, the relative position of another word with it is defined as the difference of the index numbers of the two words in the sequence. Adjacent words usually have stronger semantic relevance.

In this study, 'Emotional Difference' and 'Position' were well established and integrated in the framework of 'Emotional-Semantic Matching'. To evaluate and validate the contributions of 'Emotional Difference' and 'Position' in the framework of 'Emotional-Semantic Matching', an 'Ablation Study' was conducted, in which one or more components were removed from the framework, and their individual contributions were evaluated. The configurations used in this study were: (1) 'Without Emotional Difference', in which 'Emotional Difference' was removed from the framework, and 'Position' was used in 'Emotional-Semantic Matching', (2) 'Without Position', in which 'Position' was removed from the framework, and 'Emotional Difference' was used in 'Emotional-Semantic Matching', and (3) 'Full Model', in which 'Emotional Difference' and 'Position' were used in 'Emotional-Semantic Matching'. The results of the evaluation in this study show that the performance of the model was improved using 'Emotional Difference' and 'Position' in 'Emotional-Semantic Matching'.

Generally speaking, a sentence contains two types of information: objective and subjective. The former is an account that is based on evidence and not opinions. In contrast, the latter expresses personal viewpoints, which can be of different kinds and may be opinionated or objective. Texts generated online based on users' comments and interactions often contain rich subjective emotional infor-

mation. Extracting these emotional attributes from the text can aid in gaining a deeper understanding of the semantic information expressed by these sentences. The emotional orientation features in sentences are vital in analyzing and interpreting sentence semantics and are beneficial for aligning text with sentences. However, the emotional information that is subjective and taken from such paragraph-level emotional tendencies does not specify the topic. Thus, performing a more detailed aspect-level sentiment analysis in sentences is crucial for extracting more specific emotional features. This approach aims to more comprehensively and deeply reflect a sentence's semantic information and better accomplish matching text with sentences. Drawing on the characteristics of sentence-matching tasks, this article considers extracting emotional orientation features from two perspectives: utilizing emotional differences across sentences and leveraging emotional differences within sentences.

Consider utilizing emotional differences across sentences, focusing on word interaction between sentences. Firstly, drawing on human reading habits, this article proposes a Gaussian distribution hypothesis that reflects the semantic dependence of words. Then, aspect-level emotional differences are used as prior knowledge to modify this distribution, making it more in line with the dependency relationships in the emotional space. Finally, this article proposes a focus mechanism utilizing the spread of aspect-level emotional variations, introducing
information on emotional differences in cross-sentence analysis.

The initial dot product focus mechanism is defined as:

$$\text{sore}\,(s_t, h_t) = s_t^T h_i \tag{22}$$

In the original dot product attention mechanism, at the beginning of the dot product method, what happens to the sentence x when the attention weight is between two words x_i and x_j in the sentence is as follows:

$$\text{Comp}_{i,j} = \text{Softmax}\left(x_i, x_j\right) \tag{23}$$

Subsequently, the attention vector is derived through computing:

$$\tilde{x}_i = \sum_j \text{Comp}_{i,j}\, x_j \tag{24}$$

This article proposes a method for understanding sentences that integrates multiple features to address the limitations of the dot product attention mechanism in capturing semantic dependencies without considering emotional and positional differences between words. In daily reading habits, readers assimilate various information features to comprehend sentences. Positional information between words is crucial for semantic understanding, and emotional differences influence subjective opinions expressed in sentences. Specifically, words with similar emotional tendencies are more likely to convey similar semantics.

This approach adopts a projection pattern in the word vector encoding space to enhance the expression of these dependency relationships, projecting words into an emotional distribution space based on aspect-level emotional features. In this emotional space, the positions of words indicate their emotional polarity. Words closer together in emotional space are more likely to express similar semantics. For instance, in Fig. 1, centered around word_2, word_1, which is near the central word, is more likely to share semantic similarity. At the same time, word_3, which is farther away, is unlikely to exhibit semantic similarity with the central word. Moreover, because the dot product focus mechanism neglects the positional relationship between words when calculating attention weights, it fails to capture the locally structured positional information within a sentence.

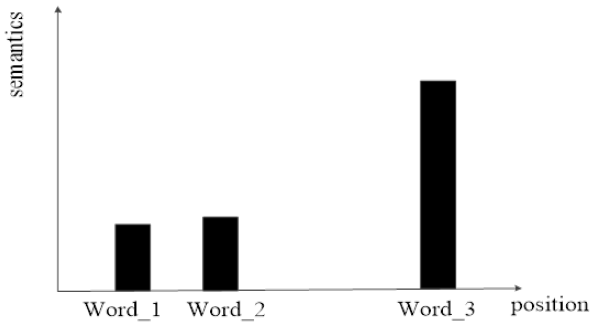

Fig. 6. Link between word distance and semantic similarity possibility

To incorporate aspect-level emotional differences into calculating attention weights, it's essential to integrate dependency information from neighboring words in the emotional space. This approach aims to enhance the original attention mechanism by considering emotional differences across words in the same sentence. This paper proposes an improved attention mechanism that adjusts the dot product attention using a Gaussian distribution to account for semantic impacts based on positional distances and emotional differences. The method begins by selecting a central word and assumes that the effect of words at diverse locations in the emotional space follows a Gaussian distribution. Leveraging aspect-level emotional differences as prior knowledge, this mechanism corrects the weights of semantic impacts from other words on the central word.

Its probability density function is $\phi(d) = e^{-\pi d^2}$, where d implies a random variable representing emotional differences between words. The variance of the standard Gaussian distribution is $\sigma^2 = 1/(2\pi)$. A sentiment classification dictionary can be trained using movie review data to incorporate aspect-level emotional differences into calculating attention weights. This dictionary allows for analyzing sentence-level sentiment features and calculating emotional differences between words. Next, this emotional difference function is integrated into the initial dot product attention mechanism to revise the impact of words at diverse locations in the emotional space on semantics. This correction introduces aspect-level emotional variation information into the attention weight computation process:

$$\begin{aligned} \tilde{x}_i &= \sum_j \frac{\phi\left(d_{i,j}\right) \operatorname{com} P_{i,j}}{Z_1} x_j \\ &= \sum_j \frac{e^{-d_{i,j}^2} e^{\left(x_i \cdot x_j\right)}}{Z_1} x_j \\ &= \sum_j \frac{e^{-d_{i,j}^2 + \left(x_i \cdot x_j\right)}}{Z_2} x_j \end{aligned} \tag{25}$$

Among them, $Z_1 = \sum_k \phi\left(d_{i,k}\right) comp_{i,k}$ and $Z_2 = \sum_k e^{-d_{i,k}^2 + (x_i \cdot x_k)}$ are regularization factors. The equation is used to change a normal distribution into a Gaussian distribution using the bias term method that allows more product operations to be involved according to various levels of simplification. However, if the variance of the Gaussian distribution misaligns with a standard normal distribution, then the necessity to introduce one more variable w emerges to ensure that the formula does not lose its effect.

$$\tilde{x}_i = \sum_j^{\sum_{i,j_{i_j}}^2} \text{Softmax} \tag{26}$$

This paper proposes introducing an emotional spatial location awareness vector into the pattern to model aspect-level emotional differences and positional information between words within the same sentence. Building on the Gaussian distribution introduced earlier, this module constructs sentiment spatial location awareness vectors utilizing the presumption of a Gaussian distribution. This approach enables the model to capture interactive dependency effects from aspect-level emotional disparities among sentence words. The computational framework for sentiment spatial location awareness dependency, leveraging the Gaussian distribution assumption, is:

$$\text{Gaussian}(n) = \exp\left(\frac{-n^2}{2\gamma^2}\right) \tag{27}$$

Among them, n displays the aspect-level emotional differences of words, and γ displays the positional information of words. During the model training process, preprocessing the statement data involves representing each word in the statement as a vector. Consequently, a matrix can be employed to depict the perceptual dependency of emotional spatial positions across diverse dimensions of word vectors:

$$P(i,\gamma) = N\left(\text{Gaussian}(n), \sigma'\right) \tag{28}$$

Here, N denotes the Gaussian distribution, σ' signifies the standard deviation, and $p(i,\gamma)$ displays the reliant influence of the i-th dimension of the word at position γ. Matrix P is the sentiment spatial position perception influence - the effect is the whole sentence. The influence vector of a particular word is represented within each column in a dimension of the matrix P, while the vector representation $P_j \in R^{d_p}$ of each word in the sentence is associated with these vectors. This perfect frame clarifies the relationship between words based on the emotive spatial positional perception.

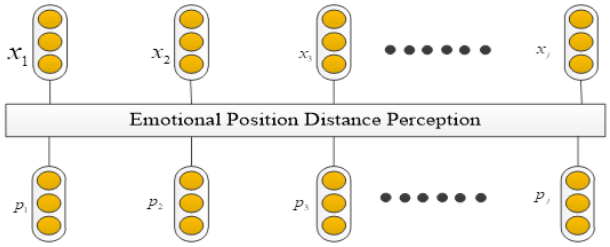

Fig. 7. Emotional Space Position Perception Vector

Thus, the sentiment spatial position perception vector of the Korean-translated text is obtained, and it is used as input for the user preference model in the following text to correct the sentiment-semantic matching of the Korean-translated text through user preferences.

4. results and discussion

4.1. Experimental Design

(1) Experimental Dataset

In this experiment, Lcqmc (large-scale Chinese question matching corpus) and BQ(Bank Question) intelligent financial customer service question matching dataset are used for model training and evaluation. The LCQMC data set contains 260068 text pairs, including 238766 training sets, 8802 development sets, and 12500 test sets. BQ data set

contains 120,000 text pairs, including 100,000 training sets, 10,000 verification sets, and 10,000 test sets. Each piece of data consists of two texts and a binary label: 0 indicates that the two texts are semantically mismatched, and 1 indicates that they are semantically matched.

It should be noted that the datasets used in this study are all pure text corpora and do not contain images, audio, or other multimodal data. The construction and use of data sets follow the common settings of semantic matching tasks in the field of natural language processing, and are suitable for the emotional semantic matching method based on user preferences and freshness proposed in this paper. In addition, Korean-English translation pairs from OPUS Parallel Corpus (http://opus.nlpl.eu/) are used for feature extraction of Korean translation texts. This corpus is an open multilingual parallel corpus resource, which is widely used in machine translation and cross-language semantic matching research.

The OPUS Korean-English Parallel Corpus is used in feature extraction, whereby there are pairs of high-quality text data in different languages, e.g., Korean and English. This corpus is essential in the training of the emotional semantic matching feature, especially in the translation of Korean texts. The Korean text is split into individual words, whereby each word is represented as a high-dimensional vector based on the Word2Vec Model, whereby the Skip-

Gram method is used to represent the semantic and syntactic meaning of the words based on the context. This plays a vital role in the representation of the context of the Korean text in the subsequent emotional semantic matching feature. In addition, the Korean-English parallel corpus is used in training the model, whereby the Korean text is translated into English, ensuring that the translated text represents the emotional context of the Korean text during translation.

(2) Experimental Environment Settings

This chapter's experiment is mainly based on the Python language and the DL framework PyTorch. Table 1 displays the specific environment settings.

Software and script: DataLoader class of PyTorch 1.11.0 is used for data loading, and num_workers=8 is set to use multi-process parallel data loading to minimize the waiting time of data transmission from CPU to GPU. The data preprocessing script is based on KoNLPy (for Korean word segmentation) and the Hugging Face Transformers library (for BERT word vectorization).

(3) Large-scale data processing flow

Large-scale data processing flow is as follows:

Block reading and streaming: 100 GB of data cannot be loaded into memory at one time. A user-defined data iterator is written to read the large-scale Korean text data (in. txt or. csv format) stored on the hard disk in chunks, and only one batch of data is loaded into the memory for processing at a time.

The semantic matching time reported includes the whole pipeline of the process. This includes a number of steps involved in the process. These steps are I/O operations, tokenization, BERT encoding, and finally the matching step. In I/O operations, the whole process of reading the data from the disk and loading the data into the memory is included. In tokenization, the text data is divided and further processed. In BERT encoding, the text data represented as tokens is encoded into a high-dimensional space using a pre-trained BERT model. Finally, in the matching step, the semantic similarity between the source and target text is calculated.

Pretreatment and vectorization: after loading each batch, Korean word segmentation and BERT word vector extraction are carried out immediately. The processed vectorized data is directly sent to the GPU for calculation, and then the memory occupied by this batch is released, and the next batch is loaded.

Indexing mechanism: a memory index is established for all data files, and the position offset of each data sample on the hard disk is recorded. This enables Data Loader to quickly and randomly access any batch of data without traversing the whole file in sequence, which greatly improves the data I/O efficiency.

Model reasoning configuration: In order to accurately measure the matching time, the model is set to reasoning mode (model.eval ()), and the automatic mixing accuracy of CUDA graphics capture and PyTorch is enabled to speed up GPU calculation. The Batch Size is set to 128, which is determined after many experiments, and can achieve the highest throughput under the memory capacity of RTX 3090.

4.2. Model Training

This chapter primarily utilizes the PyTorch framework to construct and train the model. The cross-entropy loss function employed is included in the torch. The nn toolkit is responsible for calculating the model's loss. The parameters are then optimized and updated using the Adam optimizer from torch. optim package. Given the model's size and the dataset's relatively small size, the experiment is trained for only three epochs to mitigate overfitting. However, due to the limited training rounds, saving model weights after each epoch may not yield the optimal model weights. Therefore, model evaluation on the development set is conducted every 100 batches during training to refine and ob-

Table 1. Experimental environment settings

Software and Hardware	Configuration
Operating system	Ubuntu 20.04.4 LTS
Advancement tools and languages	Pycharm and Python
DL framework	PyTorch 1.11.0
GPU	NVIDIA GeForce RTX 3090 24G
CPU	Intel® Core ^{TM} i9-9900K CPU @ 3.60GHz × 16
Memory	64G

tain the optimal weights. Once trained, the optimal model weights are applied to the test set for final evaluation. Loss changes on the LCQMC and BQ training sets during model training are depicted in Figs. 8 and 9, respectively.

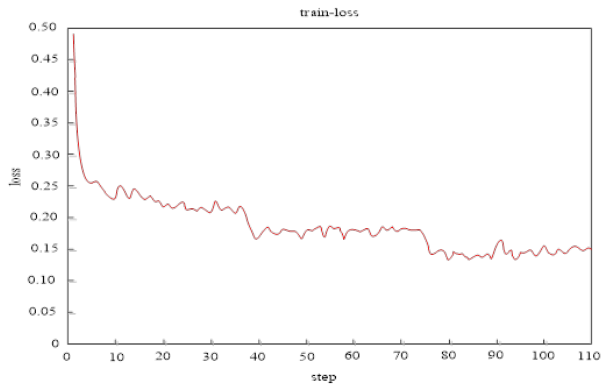

Fig. 8. Changes in loss of LCQMC training set

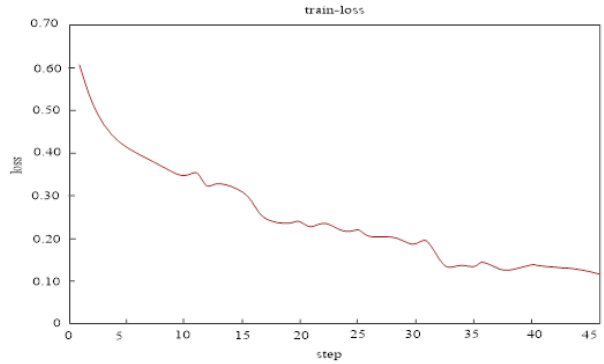

Fig. 9. Changes in BQ training set loss

Fig. 8 presents a monotone decreasing and fast convergence pattern. In the initial stage, the loss value drops sharply from the high point, which indicates that the model has been adjusted quickly and effectively through back propagation and Adam optimizer in the initial stage, and the loss curve tends to be flat and stable at a very low level in the subsequent stationary convergence period, which indicates that the model has successfully mastered the semantic matching law in LCQMC data set. The whole curve shows no signs of over-fitting, and the training loss does not fluctuate sharply, which proves the strength of the BERT-DPCNN feature extraction module and lays a solid foundation for subsequent emotional semantic matching and user preference correction.

Fig. 9(BQ training set loss change chart) shows a healthy and stable downward trend, but compared with LCQMC, its convergence speed is slightly slower, and the curve has slight and reasonable fluctuations. In the period of rapid decline and fluctuation, the overall trend of loss value is downward, but there is obvious "sawtooth" fluctuation. This is because BQ Corpus is a professional data set in the financial field, and its terminology and semantic logic are more complicated, so the model needs more time to learn the subtle differences in these specific fields. After this period of fluctuation learning, the curve finally converges to a low point smoothly. This small fluctuation is normal and healthy when training complex models to deal with complex data, which reflects the process of the optimizer actively exploring the optimal solution in the rugged "loss function terrain" and proves that the model has not fallen into mediocre local optimization. The model can successfully converge on more challenging financial data, which strongly proves the robustness and generalization ability of your proposed method.

In this research, the optimizer used is Adam. The configuration of the training process is based on certain main hyperparameters, e.g.,

- The selection of the learning rate, e.g., 0.001 .
- The selection of weight decay, e.g., 0.01 .
- The selection of batch size, e.g., 32, for the training process.

Submission Template to Journal of Applied Science and Engineering

- The selection of epochs, e.g., 3, for the training process.
- The selection of other hyperparameters, e.g., dropout, etc.

The above was based on initial experiments and fine-tuning of these hyperparameters to ensure convergence and good performance.

Feature extraction from the preprocessed data was carried out using only the OPUS Korean-English Parallel Corpus. The OPUS Korean-English Parallel Corpus contributed significantly to the extraction of vital cross-lingual linguistic features from the Korean language corpus, which facilitated the successful training of the machine learning models.

However, it is essential to note that the OPUS Korean-English Parallel Corpus played no role in the evaluation phase of the experiments. To evaluate our proposed semantic matching model, we relied on the LCQMC and BQ datasets.

OPUS Korean-English Parallel Corpus: Used exclusively in the feature extraction phase for linguistic representation learning.

LCQMC and BQ: Utilized in the evaluation phase for labeled dataset generation.

4.3. Experimental Outcomes

4.3.1. Semantic matching precision

To verify the semantic matching effectiveness of the Korean translation text using the method suggested in this article, comparisons were made with the method proposed by Batbaatar et al. and the method proposed by Pieroni et al. Semantic matching accuracy refers to the ability of the system to accurately understand and convey the emotional color of the original text and match it with the user's emotional preference in the process of Korean text translation. This requires that the translation system can not only accurately translate the literal meaning of the text, but also capture and transmit the emotional information in the text to ensure that the translation results meet the emotional expectations of users. The measurement method is to construct a dictionary containing common emotional words and their emotional polarity (positive, negative and neutral) and emotional intensity, analyze the original text and translation results, extract the emotional features, and calculate the semantic matching accuracy by using the emotional dictionary and the extracted emotional features. The formula is:

$$\text{Precision } = \frac{\mathrm{P}_1}{\mathrm{P}_2} \times 100\% \tag{29}$$

Among them, P_1 represents the number of samples that really match, which is the number of samples that are predicted to match and actually do match; P_2 represents the number of samples predicted to be matched, which is the number of all samples predicted to be matched, regardless of whether these predictions are correct or not. According to formula (25), the semantic matching accuracy of the Korean translation text of the three methods is calculated, the outcomes are displayed in Table 2.

Table 2 shows that with a Korean text volume of 100 GB , the semantic matching precision of the Korean translation text using the method proposed by Batbaatar et al is 68.2%, the method proposed by Pieroni et al is 72.8%, and that of this paper is 99.2%. With a Korean text volume of 300 GB , the precision using the method proposed by Batbaatar et al is 72.6%, the method proposed by Pieroni et al is 69.5%, and this paper's method is 99.5%. With a Korean text volume of 600 GB , the precision using the method proposed by Batbaatar et al is 72.9%, the method proposed by Pieroni et al is 69.0%, and this paper's method is 98.5%. These outcomes demonstrate that the suggested framework significantly enhances the semantic matching precision of Korean translation texts.

4.3.2. Semantic matching time consumption

The efficiency of semantic matching in Korean-translated text was evaluated using the method suggested in this paper, and comparisons were made with frameworks from the method proposed by Batbaatar et al and the method proposed by Pieroni et al, including the semantic matching times. The outcomes are displayed in Table 3.

According to Table 3, when the volume of Korean text is 100 GB , the semantic matching time for Korean translation text using the method proposed by Batbaatar et al is 11.6 s, using the method proposed by Pieroni et al is 10.5 s , and using the method suggested in this paper is 0.6s. For 300 GB of Korean text, the semantic matching times are 18.9 s, 22.8 s, and 1.9 s , respectively. When the volume increases to 600 GB , the times are 32.6 s, 42.5 s, and 3.6 s . The method suggested in this article effectively reduces matching time and enhances semantic matching efficiency.

5. conclusions

This paper puts forward the emotional and semantic matching of Korean text translation considering user preference model and freshness, which is of inestimable value for improving translation quality, enhancing user experience and promoting cross-cultural communication. Through the comprehensive application of user preference model, freshness evaluation and emotional semantic matching technology, Korean text translation can not only achieve accurate language conversion, but also cross the boundaries between culture and emotion to achieve real communication and understanding. Through experiments, we can know that:

Table 2. Semantic matching precision of Korean translation text

Korean text volume/GB	Semantic matching precision of Korean translation text/%		
	The method proposed by Batbaatar et al	The method proposed by Pieroni et al	The suggested framework
100	68.2	72.8	99.2
200	69.5	70.6	98.6
300	72.6	69.5	99.5
400	75.1	68.5	99.6
500	76.3	66.9	99.2
600	72.9	69.0	98.5

Table 3. Semantic matching time consumption

Korean text volume/GB	Korean translation text semantic matching time/s		
	The method proposed by Batbaatar et al	The method proposed by Pieroni et al	The suggested framework
100	11.6	10.5	0.6
200	16.9	15.9	1.2
300	18.9	22.8	1.9
400	22.8	32.6	2.3
500	28.5	38.1	2.9
600	32.6	42.5	3.6

(1) The semantic matching accuracy of Korean translation text based on this method can reach 99.6%;

(2) The semantic matching of Korean translation text takes 3.6s under this method;

(3) This method can effectively improve the matching accuracy and semantic matching efficiency.

Looking forward to the future, the field of Korean text translation will usher in unprecedented changes. With the continuous progress of artificial intelligence, big data and natural language processing technology, the emotion semantic matching technology considering user preference model and freshness will become more mature and popular. The future translation system will no longer be just a simple tool for language conversion, but an intelligent partner who can understand the deep-seated needs of users, keep up with the trend of the times and accurately convey emotional colors.

(1) In terms of personalized translation, the future system will be able to provide tailormade translation experience for each user through more detailed user portrait and preference analysis. Whether it is the rigor of academic research, the professionalism of business communication, or the ease and naturalness of daily conversation, it can be properly translated and presented. This highly personalized translation service will greatly enhance users' satisfaction and loyalty.

(2) In terms of freshness maintenance, the translation system will be able to track the hot topics, popular trends and emerging words around the world in real time to ensure that the translated content always keeps pace with the times. Through intelligent data mining and prediction algorithms, the system can predict in advance and integrate into the future language changes, so that the translation results will always remain cutting-edge and innovative.

(3) In terms of emotional semantic matching, with the in-depth development of emotional analysis technology, the translation system will be able to capture and convey the emotional color in the original text more delicately. Whether it is joy, sadness, anger or surprise, it can be accurately translated and expressed. This kind of emotional resonance and understanding will greatly promote the emotional communication and spiritual communication between people in different cultural backgrounds.

Competing interests

The investigators assert no competing interests.

Authorship contribution statement

Lu Cong: Project administration, Supervision, Conceptualization, Writing-Original draft preparation.

Data availability

Available upon request.

Declarations

Inapplicable.

Conflicts of interest

The investigators assert no conflicts of interest regarding this work.

Author statement

All investigators have examined and authorized the manuscript, fulfilling the criteria for authorship. Each investigator certifies that it reflects honest work.

Funding

Doctoral research start-up Fund project of Liaoning University of International Business and Economics "Study on the construction of Korean Cultural and Educational Content System from the perspective of cross-cultural Communication" (2022XJLXBSJJ06).

Ethical approval

Each author has actively and personally contributed significantly to the paper's development, and they will all be held accountable for its content.

References

[1] T. Yang, H. Wu, B. Yi, G. Feng, and X. Zhang, (2023) *"Semantic-preserving linguistic steganography by pivot translation and semantic-aware bins coding"* **IEEE Transactions on Dependable and Secure Computing *21***(1): 139–152. DOI: 10.1109/TDSC.2023.3247493.

[2] M. M. C. Eddine, (2021) *"A new concept of electronic text based on semantic coding system for machine translation"* **Transactions on Asian and Low-Resource Language Information Processing *21***(1): 1–16. DOI: https://doi.org/10.1145/3469655.

[3] R. Duan, Y. Feng, and C.-Y. Wen, (2022) *"Deep pose graph-matching-based loop closure detection for semantic visual SLAM"* **Sustainability *14***(19): 11864. DOI: https://doi.org/10.3390/su141911864.

[4] M. Kayed, F. Azzam, H. Ali, and A. Ali, (2024) *"Temporal dynamics of user activities: deep learning strategies and mathematical modeling for long-term and short-term profiling"* **Scientific Reports *14***(1): 14498. DOI: https://doi.org/10.1038/s41598-024-64120-6.

[5] W. Li, Y. Bai, and L. Meng. "Design of intelligent push system for teaching resources based on user profile". In: *2023 4th International Conference on Computer Vision, Image and Deep Learning (CVIDL)*. IEEE. 2023, 574–577. DOI: 10.1109/CVIDL58838.2023.10166991.

[6] G. Couairon, A. Grechka, J. Verbeek, H. Schwenk, and M. Cord. "Flexit: Towards flexible semantic image translation". In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2022, 18270–18279.

[7] E. Batbaatar, M. Li, and K. H. Ryu, (2019) *"Semantic-emotion neural network for emotion recognition from text"* **IEEE access 7**: 111866–111878. DOI: 10.1109/ACCESS.2019.2934529.

[8] A. Pieroni, N. Scarpato, and L. Felli, (2020) *"Blockchain and IoT convergence—a systematic survey on technologies, protocols and security"* **Applied Sciences *10***(19): 6749. DOI: https://doi.org/10.3390/app10196749.

[9] H. Yu, J. Bae, J. Choi, and H. Kim, (2021) *"LUX: smart mirror with sentiment analysis for mental comfort"* **Sensors *21***(9): 3092. DOI: https://doi.org/10.3390/s21093092.

[10] J. Zhang, H. Han, H. Su, Z. Jiang, and C. Peng, (2022) *"PUMTD: Privacy-preserving user-profile matching protocol in social networks"* **China Communications *19***(6): 77–90. DOI: 10.23919/JCC.2022.06.007.

[11] Y. Cheng and L. Kuang. "CSRS: Code search with relevance matching and semantic matching". In: *Proceedings of the 30th IEEE/ACM International Conference on Program Comprehension*. 2022, 533–542. DOI: https://doi.org/10.1145/3524610.35278.

[12] H. A. Uymaz and S. K. Metin, (2022) *"Vector based sentiment and emotion analysis from text: A survey"* **Engineering Applications of Artificial Intelligence *113***: 104922. DOI: https://doi.org/https://doi.org/10.1016/j.engappai.2022.104922..

[13] W. Ahmad, H. U. Khan, T. Iqbal, M. A. Khan, U. Tariq, and J.-h. Cha, (2023) *"Hybrid multichannel-based deep models using deep features for feature-oriented sentiment analysis"* **Sustainability *15***(9): 7213. DOI: https://doi.org/10.3390/su15097213.

[14] Z. Liu, H. Liao, M. Li, Q. Yang, and F. Meng, (2023) *"A deep learning-based sentiment analysis approach for online product ranking with probabilistic linguistic term sets"* **IEEE Transactions on Engineering Management *71***: 6677–6694. DOI: https://doi.org/10.1109/TEM.2023.3271597..

[15] P. Park, S. Jang, Y. Cho, and Y. Kim, (2024) *"SAM: cross-modal semantic alignments module for image-text retrieval"* **Multimedia Tools and Applications *83***(4): 12363–12377. DOI: https://doi.org/10.1007/s11042-023-15798-9..

[16] P. Jiang and X. Cai, (2024) *"A survey of text-matching techniques"* **Information *15***(6): 332. DOI: https://doi.org/10.3390/info15060332.

[17] S. Wang, S. Moon, I. Eum, D. Hwang, and J. Kim, (2025) *"A text dataset of fire door defects for pre-delivery inspections of apartments during the construction stage"* **Data in Brief *60***: 111536. DOI: https://doi.org/10.1016/j.dib.2025.111536.

[18] S. Wang, (2025) *"Development of an automated transformer-based text analysis framework for monitoring fire door defects in buildings"* **Scientific Reports *15***(1): 43910. DOI: https://doi.org/10.1038/s41598-025-27648-9.

[19] S. Wang, (2025) *"Graph neural network–driven text classification for fire-door defect inspection in pre-completion construction"* **Scientific Reports *15***(1): 44382. DOI: https://doi.org/10.1038/s41598-025-28120-4.

[20] S. Sharma, V. Rana, and V. Kumar, (2021) *"Deep learning based semantic personalized recommendation system"* **International Journal of Information Management Data Insights *1***(2): 100028. DOI: https://doi.org/10.1016/j.jjimei.2021.100028.

[21] J. Pei, K. Zhong, Z. Yu, L. Wang, and K. Lakshmanna, (2023) *"Scene graph semantic inference for image and text matching"* **ACM Transactions on Asian and Low-Resource Language Information Processing *22***(5): 1–23. DOI: https://doi.org/10.1145/3563390.

[22] G. Xu, X. Wang, D. Wei, Y. Shao, and B. Chen, (2023) *"Bug localization with features crossing and structured semantic information matching"* **International Journal of Software Engineering and Knowledge Engineering *33***(08): 1261–1291. DOI: https://doi.org/10.1142/S0218194023500316.

[23] A. Malkova, M.-R. Amini, B. Denis, and C. Villien, (2024) *"Neural architecture search for radio map reconstruction with partially labeled data"* **Integrated Computer-Aided Engineering *31***(3): 285–305. DOI: https://doi.org/10.3233/ICA-240732.

[24] X. Pu, L. Yuan, J. Leng, T. Wu, and X. Gao, (2023) *"Lexical knowledge enhanced text matching via distilled word sense disambiguation"* **Knowledge-Based Systems *263***: 110282. DOI: https://doi.org/10.1016/j.knosys.2023.110282.

[25] R. Qureshi, M. Irfan, T. M. Gondal, S. Khan, J. Wu, M. U. Hadi, J. Heymach, X. Le, H. Yan, and T. Alam, (2023) *"AI in drug discovery and its clinical relevance"* **Heliyon *9***(7): DOI: https://doi.org/10.1016/j.heliyon.2023.e17575.

[26] F. Wang, I. Buranaut, B. Zhang, and J. Liu, (2024) *"Emotional matching model construction of the interior interface form of age-friendly housing in Jinan city examined using Kansei engineering"* **Heliyon *10***(7): DOI: https://doi.org/10.1016/j.heliyon.2024.e2912.

[27] F. Azzam, M. Kayed, and A. Ali, (2022) *"A model for generating a user dynamic profile on social media"* **Journal of King Saud University-Computer and Information Sciences *34***(10): 9132–9145. DOI: https://doi.org/10.1016/j.jksuci.2022.08.036.

[28] J. Zhang, H. Hao, and H. Su, (2022) *"SE-$P^2\mu m$: Secure and Efficient Privacy-Preserving User-Profile Matching Protocol for Social Networks"* **IEEE Transactions on Computational Social Systems *10***(6): 3295–3307. DOI: 10.1109/TCSS.2022.3200940.

[29] L. P. de Morais, R. Immich, N. F. Silva, T. C. Rosa, and V. d. C. M. Borges, (2022) *"Characterization of the Mobile User Profile Based on Sentiments and Network Usage Attributes"* **Journal of Internet Services and Applications *13***(1): 82–97. DOI: https://doi.org/10.5753/jisa.2022.2520.

[30] K. Sudhakar, B. Smail, T. S. Reddy, S. Shitharth, D. R. Tripathi, and M. Fahlevi. "Web user profile generation and discovery analysis using LSTM architecture". In: *2022 2nd International Conference on Technological Advancements in Computational Sciences (ICTACS)*. IEEE. 2022, 371–375. DOI: 10.1109/ICTACS56270.2022.9988505.

[31] S. Kiliçarslan, (2023) *"A novel nonlinear hybrid HardSReLUE activation function in transfer learning architectures for hemorrhage classification"* **Multimedia Tools and Applications *82***(4): 6345–6365. DOI: https://link.springer.com/article/10.1007/s11042-022-14313-w.

[32] Y. Duan, J. Wang, H. Ma, and Y. Sun, (2021) *"Residual convolutional graph neural network with subgraph attention pooling"* **Tsinghua Science and Technology *27***(4): 653–663. DOI: 10.26599/TST.2021.9010058.

[33] P. Nagaraju and S. Sudha. *Retraction Note: Design of a novel panoptic segmentation using multi-scale pooling model for tooth segmentation*. 2026. DOI: https://doi.org/10.1007/s00500-026-11253-7.

[34] K. Huang, T. Sui, and H. Wu, (2022) *"3D human pose estimation with multi-scale graph convolution and hierarchical body pooling"* **Multimedia systems *28***(2): 403–412. DOI: https://doi.org/10.1007/s00530-021-00808-3.