Research Article

Analysis of Distributed English Chunk Recognition under Correlation Evaluation Based on Deep Belief Network Model

Qing Liu and Jing Zhou

School of Foreign Language, Yuzhang Normal University, Jiangxi 330003, China

Correspondence should be addressed to Qing Liu; liuqing@yuznu.edu.cn

Received 12 January 2022; Accepted 14 February 2022; Published 15 March 2022

Academic Editor: Zhiguo Qu

Copyright © 2022 Qing Liu and Jing Zhou. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Most of the traditional methods of English text chunk recognition are solved by setting the corresponding phrase identifier numbers and eventually transforming the chunk recognition problem into a lexical annotation problem. In language recognition, the traditional MFCC features are easily contaminated by noise and have weak noise immunity due to the insufficient amount of information on each frame of the signal. At the same time, SDC feature extraction methods commonly used today require artificial settings in parameter selection, which increases the uncertainty of recognition results. The method of identifying English text chunks by association evaluation of central word extensions identifies English text chunks from a different perspective. It has the following features: (i) each phase is considered as a cluster with the central word as the core, and the internal composition pattern of each phrase is fully considered; (ii) the results are dynamically evaluated using association and confidence. The results show that the proposed method can achieve higher recognition rate than traditional feature extraction methods. The recognition rate is faster, and the F-measure value of English block recognition reaches 94.05%, which is comparable to the best results so far.

1. Introduction

Block recognition is the main element of shallow analysis, which can be applied to information retrieval, machine translation, subject content analysis, and text processing, and the accuracy of block recognition is directly related to the correctness of text analysis and text processing. Since [1] proposed a strategy for shallow syntactic analysis and designed and implemented a discourse block recognizer, shallow syntactic analysis has received general attention, and the theme of the 2018 CONLL conference was shallow syntactic analysis [2]. Public training and test set were given at this conference. Various statistical and machine learning methods were subsequently applied to English chunk recognition [3]. Subsequently, [4] used the machine learning algorithm Winnow for English chunk recognition and obtained the best results reported so far (with an accuracy rate of over 94.28%). The advantage of this algorithm is that it can identify features relevant to itself from a large number of features, but the use of a large number of features makes the query inefficient; also, the use of lexicalized features leads to sparse data [5].

An analysis of previous research methods shows that the current strategy of chunk recognition is to turn the chunk recognition problem into a classification problem similar to lexical annotation, but the disadvantage of this approach is that it cannot take into account the constituent features within each phrase [6]. In view of this, this paper extends the boundaries of a phrase by the degree of association between two adjacent lexemes, using the central word as the core; it proposes the concepts of suspicion and plausibility and then evaluates the degree of association by determining the values of suspicion and plausibility in an error-driven way and then corrects the results just obtained. The results obtained using this method are comparable to the best current results [7].
With the advent of the information age and the development of the Internet, language recognition has become increasingly valuable.

The earliest research on language recognition dates back to 1974, when TI used sequences of phonetic units to classify different languages [8]. Over the last 40 years, the development of language recognition has evolved, and the technology has matured, resulting in a mainstream approach to language identification using parallel Gaussian mixture models [9]. The Mel-Frequency Cepstral Coefficient (MFCC) feature, which is commonly used in language recognition systems today, is susceptible to noise contamination, and its noise immunity is weak as each frame usually contains only 20-30 ms of speech signal [10]. For another feature extraction method, Shifted Delta Cepstra (SDC) [11], although it is a great improvement over MFCC parameters, the parameters of SDC are artificially set, making it the parameters of SDC are artificially set, which makes it universally applicable to all speech data [12].

In this paper, a new feature extraction method, called BN, is proposed by combining bottleneck (BN) and deep belief network (DBN), which is an approximation of artificial neural networks (ANN) [13]. DBN has the advantages of less stringent requirements on the internal statistical structure and density function of the input data, the ability to process speech data over longer time periods, and greater robustness to interference, such as different speakers’ speaking styles, accents, and external noise, and therefore has stronger modeling and characterization capabilities [14].

In this paper, we conducted language recognition experiments using the bottleneck (BN) [15] and DBN methods with data from the NIST07 phonetic database. The experimental results show that the BN-DBN method can improve the recognition accuracy more effectively than the traditional language recognition methods MFCC and SDC [16].

2. English Chunk Recognition Based on Central Word Expansion

**Definition 1** (phrasal centrality). A word that occurs more often in a phrase is a central lexeme.

**Definition 2** (phrasal centrality). The word that corresponds to the central lexeme of a phrase is the central word of the phrase.

**Definition 3** (relatedness). It is a probability value that measures how closely two adjacent lexemes are related to each other.

Let the number of occurrences of word $a$ in a phrase $X$ be $ATa$, the number of occurrences of word $b$ be $ATb$, and the number of occurrences of word $b$ next to word $a$ be $AT(a \rightarrow b)$; then, the degree of association $R(a \rightarrow b)$ between word $a$ and word $b$ in a phrase of $X$ is given by

$$R(a \rightarrow b) = \frac{AT(a \rightarrow b)}{ATa} \times \frac{AT(a \leftarrow b)}{ATb}.$$  \hspace{1cm} (1)

In the above equation, $AT(a \rightarrow b)ATa$ shows the importance of $b$ for $a$ in $X$ phrase, while $AT(a \rightarrow b)ATb$ shows the importance of $a$ for $b$ in $X$ phrase, which reflects the mutual selection relationship between two lexical forms in the same phrase.

We consider phrases to be clusters of lexical properties with central lexically as the core. The top two lexical properties (central lexical properties) of each phrase were obtained by counting the training set [17].

The process of identifying English chunks based on association is as follows: starting from the central word, the association between two adjacent words is continuously calculated to both sides, and if the association is greater than a threshold, it continues to expand to both sides until it stops when the association is less than the threshold. Most of the phrases do not overlap in central lexically, but some do (e.g., PRT and SBAR and ADVP and CONJP). An overlap in the central lexeme indicates that the current lexieme is central to more than one phrase. In this case, each phrase is expanded from that central word to both sides. After the above operation, many candidate phrases are generated, and there are boundary conflicts between the candidate phrases, for which we adopt a greedy method [18].

In the abovementioned process of English chunk recognition based on central word expansion, the correlation degree plays a decisive role, but it is not ideal to rely solely on the correlation degree obtained from the training set to identify chunks, mainly because the correlation degree is static and cannot adapt to the complex situation in the chunk recognition process. In view of this, we have developed a mechanism to evaluate the correlation degree by using the skepticism and the trustworthiness to evaluate the original correlation degree and reevaluate the correlation degree.

**Definition 4** (suspicion). It is a probability value that measures the likelihood of a lexical correlation being incorrect in the recognition process.

**Definition 5** (credibility). It is a probability value that measures the likelihood that a lexical correlation is correctly marked in the recognition process. The degree of suspicion is set to prevent errors in the selected central word. For example, if a word is seen as the central word in an utterance, and the word is highly related to the words around it, expanding the boundaries of the phrase around it; but it is not really the central word itself but just a common word in the phrase in which it is found, and it is highly related to the words around it because the words around it are also in the same phrase as it; this may cause the central word to be set away from the true centre and thus bias. This can lead to a deviation from the true centre of the word and thus bias the labeling. By adding the degree of suspicion, the degree of suspicion increases as the boundary of the misdirected central word expands to either side, and eventually, the total degree of suspicion will be greater than the suspicion threshold not far from the misdirected central word, at which point the node will be invalided, although the correlation of the lexical properties on either side of the boundary will still be
greater than the threshold. Reliability serves as a measure of the reliability of the association between lexemes [19].

In the process of recalculation the association degree, it is necessary to count the number of times each association degree produces a correct effect, the number of times it produces an incorrect effect, and the number of times it is recalled and assigns a certain weight to each of these three times for each association degree—the three weight values are called the correct rate, the incorrect rate, and the correction coefficients for the confidence and doubt of the correlations are obtained from these three rates. The correction coefficients are given as follows:

\[
(Xa \leftarrow b) = \frac{(X, a \leftarrow b)}{AT(X, a \leftarrow b)} \times \text{AgreeRatio} \\
\frac{(X, a \leftarrow b)}{AT(X, a \leftarrow b)} + \text{ErrorRatio} \\
\frac{(X, a \leftarrow b)}{AT(X, a \leftarrow b)} + \text{RecallRatio} \\
\times \text{RecallTmes} \\
\times \text{ErrorTmes} \\
\times \text{AgreeTmes} \\
\]

where AgreeRatio is the correct rate, ErrorRatio is the error rate, and RecallRatio is the recall rate; these values are set in Table 1; AgreeTimes is the number of correct, ErrorTimes is the number of errors, and RecallTimes is the number of recalls.

New credibility = original credibility × (1 + correction factor) - (initial value of credibility = 1),

New skepticism = (-1) × original skepticism × (1 + correction factor) (first correction of skepticism = correction factor).

The new correlation is calculated as follows:

\[
(Xa \leftarrow b) = \frac{(X, a \leftarrow b)}{AT(X, a \leftarrow b)} + AT(Xa \leftarrow b). \tag{4}
\]

TrustDegree (X : a → b) is the trustworthiness of the association of the lexical property b with a, with X denoting the phrase type. The calculation of the new association is actually an error-driven process, in which the annotation result is compared with the correct answer, causing the trustworthiness of the annotated association to increase and the skepticism to decrease for the correct effect, and conversely, causing the trustworthiness of the annotated association to decrease and the skepticism to increase for the incorrect effect. For skepticism, a wrongly chosen central word will cause the skepticism between it and the surrounding words to rise, thus speeding up the rise of the total skepticism in the labeled nodes and allowing the wrong choice of the presumed central word to be exposed as soon as possible [10].

When the results of the block identification are no longer rising, the final values of correlation, suspicion, and confidence are obtained and applied to the open test.

3. BN-DBN for Speech Feature Extraction

DBN, formally proposed by [20], is theoretically a method for learning models with deep structure (i.e., containing multiple layers of nonlinear arithmetic units), and it has a stronger modelling and characterization capability when dealing with real-world data (e.g., natural speech, natural images, and video) than previous modelling methods for “shallow” structure (i.e., containing only a single layer of nonlinear arithmetic units). The DBN is still essentially a modeling and characterization method for real-world data (e.g., natural speech, natural images, and video). DBN is still essentially a multilayer ANN, but it uses a combination of supervised and unsupervised training to obtain the network parameters, solving the problem that the ANN back propagation algorithm can easily fall into a local optimum [21].

The concept of bottleneck was first introduced by [22] and applied to continuous speech recognition, while BN-DBN is the result of combining the concept of bottleneck with DBN. The BN-DBN is usually set up as a multilayer ANN with an odd number of layers, and the middle layer is named as the bottleneck layer [23]. As the name implies, bottleneck means that the number of neurons in the layer is much smaller than the other layers. The BN-DBN-based approach to speech feature extraction can be implemented in two steps.

Step 1. Construct a nerve network and build a DBN through pretraining and fine-tuning.

Compositionally, a DBN is a series of restricted Boltzmann machine (RBM) cascades, and the composition of a complete DBN is shown in Figure 1.

As shown in the figure, an RBM consists of a visible layer \(v_i\) and a hidden layer cell \(h_j\) interconnected, and the joint distribution for a given set of model parameters can be expressed as an energy function:

\[
E(v_i, h_j; \theta) = -\sum_{ij} w_{ij}v_i h_j - \sum_i b_i v_i - \sum_j a_j h_j,
\]

\[
E(v_i, h_j; \theta) = -\sum_{ij} w_{ij}v_i h_j - \sum_i b_i v_i - \sum_j a_j h_j, \tag{5}
\]

where \(\theta = \{w, a, b\}\) and \(w_{ij}\) are the connection weights of the visible and hidden units. \(b_i\) and \(a_j\) are the corresponding biases, respectively. The probability density distribution can be determined using the Boltzmann distribution:

\[
P_q(v, h) = \frac{1}{Z(\theta)} \exp (-E(v, h; \theta)) \\
= \frac{1}{Z(\theta)} \prod_{ij} e^{v_i h_j} \prod_i e^{b_i v_i} \prod_j e^{a_j h_j}, \tag{6}
\]
where \( Z(\theta) = \sum_{h,v} \exp(-E(v,h;\theta)) \), because the hidden nodes are conditionally independent of each other, i.e.,

\[
P(h|h) = \prod_j P(h_j|h),
\]

\[
P(v|h) = \prod_j P(v_j|h),
\]

\[
P(v=1|h) = \frac{1}{1 + \exp(-\sum_j w_{ij}h_j - b_j)}.
\]

Maximise the following log-likelihood functions:

\[
L(\theta) = \frac{1}{N} \sum_{n=1}^{N} \log P_y(v) - \frac{\lambda}{N} \|w\|^2.
\]

The derivative of the maximum log-likelihood function yields the parameter \( w \) corresponding to the maximum \( L \).

\[
\frac{\partial L(\theta)}{\partial w_{ij}} = E_{P_{rus}}[v_ih_j] - E_{P_{ru}}[v_ih_j] - \frac{2\lambda}{N} w_{ij}.
\]

We can use a supervised learning approach similar to that of a traditional BP neural network to build the entire DBN can then be built using a supervised learning approach similar to that of a traditional BP neural network, with back-to-back callbacks [24]. For Step 2, as showed in Figure 2, the network after the bottleneck layer is removed, and the original bottleneck layer is used as the output layer.

### 4. Analysis and Comparison of Experimental Results

#### 4.1. Relevance Evaluation Analysis.

We used the common training (WSJ15-18) and test sets (WSJ20). After testing, the best results were obtained by selecting the thresholds and rates in Table 2.

| Relative threshold | Doubt threshold | AgreeRatio | ErrorRatio | RecallRatio | Precision (%) | Recall (%) | Training times |
|--------------------|----------------|------------|------------|-------------|---------------|------------|----------------|
| 0.01               | 0.95           | 0.1        | 0.1        | 0.05        | 79.55         | 86.62      | 1              |
| 0.01               | 0.95           | 0.1        | 0.1        | 0.05        | 80.01         | 86.69      | 2              |
| 0.01               | 0.95           | 0.1        | 0.1        | 0.05        | 81.48         | 87.00      | 3              |
| 0.01               | 0.95           | 0.1        | 0.1        | 0.05        | 82.57         | 87.73      | 4              |
| 0.01               | 0.95           | 0.1        | 0.1        | 0.05        | 83.09         | 87.98      | 5              |
| 0.01               | 0.95           | 0.1        | 0.1        | 0.05        | 84.42         | 88.25      | 6              |
| 0.01               | 0.95           | 0.1        | 0.1        | 0.05        | 84.45         | 88.90      | 7              |
| 0.01               | 0.95           | 0.1        | 0.1        | 0.05        | 85.45         | 89.15      | 8              |
| 0.01               | 0.95           | 0.1        | 0.1        | 0.05        | 86.12         | 90.10      | 9              |
| 0.01               | 0.95           | 0.1        | 0.1        | 0.05        | 87.66         | 91.08      | 10             |
sessions, some confidence levels reach above 2.0, while some doubt levels are even below -1.0. Therefore, the number of training sessions should be 5 to 7 [25].

The reasons for setting the three training rates at 0.3, 0.2, and 0.1 are as follows: if a value greater than these three numbers is chosen, the correction process of the rule model will become chaotic due to the large changes in each time, while a smaller value will achieve similar results, but the training time will be longer. Table 1 shows the correct rate, error rate, and recall rate with 0.1 and 0.05, respectively. Comparing Tables 1 and 2, it is clear that the number of training sessions in Table 1 is significantly higher than the number of sessions in Table 2 to achieve the best results.

The first three methods in Table 3 are the current leading methods in the field of English chunk recognition results (11 phrases recognized on the same common training and test sets). From Table 3, it is easy to see that the results obtained by the method of central word expansion based on relevance evaluation are comparable to the current best results; although slightly lower than the Winnow-based method, this method has the following advantages over the Winnow method: (i) the Winnow method uses more features and therefore occupies a larger amount of memory, whereas this method uses fewer types and numbers of features and therefore occupies less memory. The time complexity of this algorithm is O(n), and the recognition speed is fast; from training to the end of recognition on a computer with a main frequency of 1.5 GHz, the running time is less than 5 minutes, while the training time of the Winnow method is 22 minutes [26].

4.2. Language Identification Experiments and Analysis. Experimental voice library from [25] and data from telephone recordings and realistic conversation style, containing noise, pauses, breaths, repetitions, incomplete pronunciations, accents, etc., are sampled at 8 kHz.

The number of neurons in the bottleneck layer was set to 20, 25, 30, 35, 39, 50, and 60 based on the performance comparison of BN-DBN with different BN layers, and the results are listed in Table 4.

Performance comparison based on 4 different feature extractions. These 4 different features are as follows:

| Relative threshold | Doubt threshold | AgreeRatio | ErrorRatio | RecallRatio | Precision (%) | Recall (%) | Training times |
|-------------------|----------------|------------|------------|-------------|---------------|------------|----------------|
| 0.01              | 0.95           | 0.3        | 0.2        | 0.1         | 79.49         | 86.62      | 1              |
| 0.01              | 0.95           | 0.3        | 0.2        | 0.1         | 80.2          | 87.4       | 2              |
| 0.01              | 0.95           | 0.3        | 0.2        | 0.1         | 84.64         | 88.29      | 3              |
| 0.01              | 0.95           | 0.3        | 0.2        | 0.1         | 87.69         | 91.29      | 4              |
| 0.01              | 0.95           | 0.3        | 0.2        | 0.1         | 90.75         | 93.02      | 5              |
| 0.01              | 0.95           | 0.3        | 0.2        | 0.1         | 93.87         | 94.23      | 6              |
| 0.01              | 0.95           | 0.3        | 0.2        | 0.1         | 91.55         | 93.08      | 7              |
| 0.01              | 0.95           | 0.3        | 0.2        | 0.1         | 87.73         | 90.77      | 8              |
| 0.01              | 0.95           | 0.3        | 0.2        | 0.1         | 86.74         | 88.8       | 9              |
| 0.01              | 0.95           | 0.3        | 0.2        | 0.1         | 84.76         | 86.8       | 10             |

Table 3: Comparison of the results of the method in this paper with other methods.

| Result                              | SVM       | Memory-based learning | Winnow | Headword |
|-------------------------------------|-----------|-----------------------|---------|----------|
| Precision (%)                       | 93.45     | 94.04                 | 94.28   | 93.87    |
| Recall (%)                          | 93.51     | 91.00                 | 94.07   | 94.23    |
| F1 (%)                              | 93.48     | 90.50                 | 94.17   | 94.05    |

Table 4: Performance comparison of BN-DBN based on different BN layers (EER%).

| Features    | 30 s | 10 s |
|-------------|------|------|
| BN-DBN20    | 1.91%| 6.62%|
| BN-DBN25    | 1.86%| 6.42%|
| BN-DBN30    | 1.86%| 6.41%|
| BN-DBN35    | 1.85%| 6.39%|
| BN-DBN39    | 1.82%| 3.64%|
| BN-DBN50    | 1.87%| 6.47%|
| BN-DBN60    | 1.94%| 6.61%|

Table 5: Comparison of performance based on different feature extraction (EER%).

| Features    | 30 s | 10 s |
|-------------|------|------|
| MFCC39      | 5.63%| 10.45%|
| MFCC39-11   | 2.74%| 7.28%|
| SDC         | 1.93%| 6.79%|
| BN-DBN39    | 1.82%| 6.34%|

The language samples were first passed through a pre-emphasis filter $H(z) = 1-0.97z^{-1}$ and then averaged over multiple frames, with each frame 256 points long and 128 points shifted, using a Hamming window function and a filter bank of 24 Mel triangular filters.

Based on MFCC39, each frame is extended by 5 frames (5 + 1 + 5) before and after each frame, resulting in a new 11-frame parameter [27].

Based on MFCC39, the first 7 order coefficients (C0-C6) are obtained, and MFCC is expanded by (7, 1, 3, 7)
(N, d, P, k) to 49-dimensional features, and the 49-dimensional SDC and 7th order MFCC coefficients are stitched together to obtain the 56-dimensional SDC feature parameters for final use.

The extracted MFCC39-11 features were fed into a 5-layer BN-DBN network, where the number of neurons in the 5 layers was 1024-512-39-39-512-1024 combinations. An initial DBN can be constructed by learning a large number of layer-by-layer bottom-up RBMs, and finally, the whole DBN is fine-tuned from back to front using supervised learning similar to that of traditional BP neural networks. The results are listed in Table 5.

5. Conclusions

In this paper, each phrase is viewed as a cluster of lexical properties with the central lexical property as the core. The central lexical property is selected first, and then, the boundaries of the phrase are expanded by the correlation between two adjacent lexical properties, and an error-driven method is applied to evaluate and correct the correlation to improve the accuracy of phrase recognition. This paper addresses the problems that most of the current speech feature extraction methods cannot make full use of multiframe information, are sensitive to external interference, and require more parameters to be set manually, and proposes the method of picking out bottleneck depth belief network to solve these problems, so as to achieve the ultimate goal of improving recognition accuracy. Three experiments in the NIST2007 database show that the bottleneck depth belief network algorithm achieves better recognition accuracy than the other three algorithms compared in the paper.

Data Availability

The experimental data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declared that they have no conflicts of interest regarding this work.

References

[1] Q. Tang, “Analysis of English multitext reading comprehension model based on deep belief neural network,” Computational Intelligence and Neuroscience, vol. 2021, no. 5, Article ID 5100809, p. 10, 2021.
[2] J. Wu, C. Shi, M. Shao et al., “Reactive power optimization of a distribution system based on scene matching and deep belief network,” Energies, vol. 12, no. 17, p. 3246, 2019.
[3] X. Rui and L. Yang, “A multi-task learning framework for emotion recognition using 2D continuous space,” IEEE Transactions on Affective Computing, vol. 8, no. 1, pp. 3–14, 2017.
[4] I. Chaturvedi, Y. S. Ong, I. W. Tsang, R. E. Welsh, and E. Cambria, “Learning word dependencies in text by means of a deep recurrent belief network,” Knowledge-Based Systems, vol. 108, pp. 144–154, 2016.
[5] N. A. Kallioras, G. Kazakis, and N. D. Lagaros, “Accelerated topology optimization by means of deep learning,” Structural and Multidisciplinary Optimization, vol. 62, no. 3, pp. 1185–1212, 2020.
[6] Z. Lei, B. C. Pijanowski, K. Alexandridis, and J. Olson, “Distributed modeling architecture of a multi-agent-based behavioral economic landscape (MABEL) model,” Simulation, vol. 81, no. 7, pp. 503–515, 2005.
[7] C. Zhang, J. Cheng, X. Tang, V. S. Sheng, Z. Dong, and J. Li, “Novel DDoS feature representation model combining deep belief network and canonical correlation analysis,” Computers, Materials and Continua, vol. 58, no. 2, pp. 657–675, 2019.
[8] W. Lei, C. Xinyan, and L. Maohua, “English letter recognition based on Tensor Flow deep learning,” Journal of Physics: Conference Series, vol. 1627, no. 1, article 012012, 2020.
[9] A. Lozano-Diez, R. Zazo, D. T. Toledano, and J. Gonzalez-Rodriguez, “An analysis of the influence of deep neural network (DNN) topology in bottleneck feature based language recognition,” PLoS One, vol. 12, no. 8, article e0182580, 2017.
[10] L. Zhou and Q. Zhang, “Recognition of false comments in E-commerce based on deep learning confidence network algorithm,” in Information Systems and e-Business Management, pp. 1–18, Springer, 2021.
[11] Y. Li, R. Ma, and R. Jiao, “A hybrid malicious code detection method based on deep learning,” International Journal of Software Engineering and its Applications, vol. 9, no. 5, pp. 205–216, 2015.
[12] H. J. Lee and D. Lee, “Study of process-focused assessment using an algorithm for facial expression recognition based on a deep neural network model,” Electronics, vol. 10, no. 1, p. 54, 2020.
[13] A. Alameen and A. Gupta, “Clustering and classification based real time analysis of health monitoring and risk assessment in wireless body sensor networks,” Bio-Algorithm and Med-Systems, vol. 15, no. 4, 2019.
[14] T. T. Shi, Y. B. Zhang, L. P. Guo, Z. X. Jing, and L. Q. Huang, “Research on remote sensing recognition of wild planted Lonicer japonica based on deep convolutional neural network,” Chinese Journal of Chinese Materia Medica, vol. 45, no. 23, pp. 5658–5662, 2020.
[15] J. H. Chen, Y. H. Hao, H. Wang, T. Wang, and D. W. Zheng, “Futures price prediction modeling and decision-making based on DBN deep learning,” Intelligent Data Analysis, vol. 23, pp. 53–65, 2019.
[16] J. Morris and P. J. Holcomb, “Event-related potentials to violations of inflectional verb morphology in English,” Cognitive Brain Research, vol. 25, no. 3, pp. 963–981, 2005.
[17] T. Zhao, Y. Liu, G. Huo, and X. Zhu, “A deep learning iris recognition method based on capsule network architecture,” IEEE Access, vol. 7, pp. 49691–49701, 2019.
[18] G. Gazzola, C. A. Chou, M. K. Jeong, and W. A. Chaovalitwongse, “An introduction to the analysis of functional magnetic resonance imaging data,” Fields Institute Communications, vol. 63, pp. 131–151, 2013.
[19] D. P. Corina, L. A. Lawyer, and H. Peter, “Lexical processing in deaf readers: an fMRI investigation of reading proficiency,” PLoS One, vol. 8, no. 1, article e54696, 2013.
[20] M. Bingenheimer, “The digital archive of Buddhist Temple Gazeteers and named entity recognition (NER) in classical Chinese,” Lingua Sinica, vol. 1, no. 1, p. 8, 2015.
[21] F. Li, C. Wang, X. Liu, Y. Peng, and S. Jin, “A composite model of wound segmentation based on traditional methods and deep neural networks,” *Computational Intelligence and Neuroscience*, vol. 2018, Article ID 4149103, 12 pages, 2018.

[22] L. Besacier, A. M. Ariyaeinia, J. S. Mason et al., “Voice biometrics over the Internet in the framework of COST Action 275,” *EURASIP journal on advances in signal processing*, vol. 2004, no. 4, p. 479, 2004.

[23] M. Moura and E. L. Droguett, “A continuous-time semi-Markov Bayesian belief network model for availability measure estimation of fault tolerant systems,” *Pesquisa Operacional*, vol. 28, no. 2, pp. 355–375, 2008.

[24] D. E. Bernard, “Multimodal natural language query system for processing and analyzing voice and proximity-based queries,” *Journal of the Acoustical Society of America*, vol. 130, no. 1, p. 640, 2015.

[25] N. Marir, H. Wang, G. Feng, B. Li, and M. Jia, “Distributed abnormal behavior detection approach based on deep belief network and ensemble SVM using spark,” *IEEE Access*, vol. 6, pp. 59657–59671, 2018.

[26] H. Li, D. Zeng, L. Chen, Q. Chen, M. Wang, and C. Zhang, “Immune multipath reliable transmission with fault tolerance in wireless sensor networks,” in *International Conference on Bio-Inspired Computing: Theories and Applications*, pp. 513–517, Springer, Singapore, 2016.

[27] C. H. Cao, Y. N. Tang, D. Y. Huang, G. Wei Min, and Z. Chunjong, “IBE: an improved identity-based encryption algorithm for WSN security,” *Security and Communication Networks*, vol. 2021, Article ID 8527068, 8 pages, 2021.