In order to solve the problem of multiuser security sharing and privacy protection of the speech data in cloud storage and realize efficient encrypted speech retrieval, an encrypted speech retrieval scheme based on multiuser searchable encryption was proposed. Firstly, the ciphertext-policy attribute-based encryption (CP-ABE) and searchable encryption (SE) are combined to support a multiuser searchable speech encryption scheme, which achieves the encryption and fine-grained access control of the speech data. Secondly, the Mel frequency cepstral coefficient (MFCC) feature of the original speech that is used as the input of the long- and short-term memory network (LSTM) is extracted to perform the deep semantic feature extraction as the speech keywords. Finally, the speech keywords are encrypted to generate a secure index, bound together with the encrypted speech, and then stored in the cloud. During the user retrieving, the keywords of query speech are extracted by utilizing the trained LSTM to generate the search trapdoor of the user and then uploaded to the cloud server, and the Euclidean distance is used for matching the security index with the search trapdoor. In addition, a proxy server is introduced to execute the partial ciphertext decryption operations to reduce computation overhead and storage space. The theoretical analysis and experimental results show that the proposed scheme has higher security and retrieval accuracy and can realize the secure storage of the massive speech data and multiuser data sharing.

1. Introduction

In recent years, cloud storage has become a popular storage solution, more and more enterprises and individuals tend to outsource data to the cloud to save local storage and computing costs, and users can also share their files on the cloud. However, the cloud service providers (CSPs) are not completely trusted. While the popularity and application of cloud storage bring convenience to users, they also bring challenges of security and privacy issues. Once data is stored in the cloud, users will lose their direct control over the data, which could cause sensitive information to leak, especially sensitive speech data [1, 2]. In order to ensure the privacy and confidentiality of the data, users usually encrypt the data and then store it on the cloud server in the form of ciphertext. But the encrypted speech data often loses most of its features, which reduces the searchability in massive ciphertext data and increases the difficulty of retrieving data for users [3, 4].
single-user environment, which is unable to satisfy the requirements in multiuser scenarios that users share and search the files in the remote cloud storage server. With the popularity of the conception of cloud storage and sharing, users and data have witnessed a surge of growing, and data sharing is becoming more and more important. Therefore, how to retrieve encrypted speech data efficiently with privacy-preserving for multiple users in cloud storage has become an urgent issue to be settled.

The existing works for content-based encrypted speech retrieval schemes do not consider that legitimate users can access and retrieve encrypted data safely under the control of the data owner, and it is difficult to meet the retrieval requirements in multiuser scenarios. Moreover, the CP-ABE scheme can achieve data security in the cloud and fine-grained access control to effectively prevent unauthorized access. It has become a vital access control technology for secure data sharing and multiuser searchable encryption in cloud storage.

In the existing SE schemes, there are fewer approaches for extracting keywords from the speech that needs to be stored in the cloud server. In addition, the existing SE technology is implemented by matching keywords and is not suitable for encrypted speech data with perceptual redundancy characteristics. Thanks are due to the deep learning technique that has been widely used in image recognition/retrieval [6–8], speaker recognition [9], speech recognition/retrieval, and other fields [10–13] with its powerful automatic feature learning ability and efficient feature description. This paper exploits the LSTM network model for speech deep feature extraction and recognition and uses the output as speech keyword, which not only achieves the purpose of extracting speech keywords but also has high retrieval accuracy.

The above researches are analyzed in order to ensure the safe sharing and privacy protection of the speech data in the cloud by multiple users and realize efficient multiuser encrypted speech retrieval. The CP-ABE scheme is adopted for privacy protection of speech data and multiuser access control, and the LSTM network model is employed to extract the deep semantic features of the speech data that are taken as the keywords involved in the SE scheme. Based on these, a multiuser searchable encryption speech retrieval scheme supporting one-to-many mode is presented in this paper. The main contributions of this work are as follows:

1. An encrypted speech retrieval scheme is present that supports multiuser search, which also allows data owners or legally authorized multiple users to send speech search trapdoors to retrieve encrypted speech data stored in cloud servers.
2. The combination of CP-ABE and SE schemes can not only ensure the privacy of speech data and achieve fine-grained access control of users in cloud storage but also realize the secure sharing of speech data.
3. A novel method for keyword extraction of speech content is proposed, which uses the deep features extracted by the LSTM network model as speech keywords to improve retrieval accuracy.

The rest of this paper is arranged as follows: Section 2 analyzes relevant research work in detail. Section 3 explains the related theoretical basis. Section 4 presents a multiuser encrypted speech retrieval scheme and its implementation process. Section 5 carries out the theoretical analysis and experimental verification of the proposed scheme compared with the existing method performance. Section 6 summarizes the work of our paper.

2. Related Work

The existing SE schemes have four application modes, namely, one-to-one mode (S/S), one-to-many mode (S/M), many-to-one mode (M/S), and many-to-many mode (M/M); according to application scenarios, these four modes can be divided into single-user searchable encryption and multiuser searchable encryption [14]. Song et al. [15] first proposed a single-user searchable encryption scheme based on sequential scanning. The scheme enables users to store encrypted data in the cloud and perform keyword search through the ciphertext domain. Although it is saving amounts of overhead for users, the search efficiency is low. Li et al. [16] proposed a multikeyword fuzzy searchable encryption retrieval scheme over encrypted cloud storage data. It utilizes inner product to compute relevance to achieve matching retrieval, and the locality sensitive hashing (LSH) and bloom filter are used which obtains high search accuracy. Considering the fuzzy semantic problem of keywords, Liu et al. [17] proposed a fuzzy semantic SE scheme (FSSE), which utilizes a fingerprint generation algorithm and semantic expansion technique to realize fuzzy semantic search by using the Hamming distance. Although the scheme is secure and privacy-preserving, there exists a collusion attack between two cloud servers.

However, the above schemes are all single-user searchable encryption schemes that the application scenarios are limited, and very few researchers are engaged in multiuser searchable encryption. Curtmola et al. [18] first presented a multiuser SE scheme to allow a group of users to search the owner’s files by using broadcast encryption. Since then, researchers have proposed solutions for different functions in multiuser scenarios. Ranjan et al. [19] proposed a privacy-preserving scheme that supports keyword search over encrypted data in multiple data owner and multiuser scenarios. The scheme realizes privacy protection and access control by adopting attribute encryption and improves search efficiency by tree index. Deng et al. [20] proposed an efficient multiuser SE scheme with keyword authorization in cloud storage. The solution enables owners to share and search their encrypted files with authorized users. Type-3 asymmetric bilinear mapping and keyword authorization binary tree is used to obtain better performance and prove the security under BDHV and SXDH assumptions. To settle the issue of fine-grained access control for multiuser searchable encryption, Cao et al. [21] proposed a multiuser searchable privacy-preserving scheme for medical data in cloud storage. Owners adopt the CP-ABE scheme to achieve privacy protection and fine-grained access control for...
multiple users, but the access control attribute lists have some more redundancy, and the attribute set needs further optimization. Lv et al. [22] proposed multiuser searchable encryption with efficient access control in cloud storage. To make retrieval of encrypted data for multiusers by CP-ABE technique and delegate most decryption to the proxy server, Morales-Sandoval et al. [23] gave an attribute-based searchable encryption (ABSE) scheme for secure cloud storage, sharing, and retrieval of encrypted data under the user’s access control. Aiming at the problem of secure Boolean query and ranked search in mobile cloud, Chen et al. [24] presented a multiuser Boolean keyword search scheme (MBKSS), which achieves a rapid Boolean query outcome, and access tree is utilized to access authorization, and they also designed a homomorphic cryptosystem with partial decryption to protect search privacy and rank search.

Considering the privacy and searchability of multimedia data stored in the cloud, Yu et al. [25] constructed public-key encryption with a keywords search (PEKS) model based on the lattice that supports multiuser search, where the delegate accesses the encrypted multimedia database with time constraint through proxy reencryption. Yang et al. [26] proposed a data retrieval scheme for multiple users based on the lattice assumption in the multimedia cloud. The scheme uses broadcast encryption to ensure that encrypted multimedia data can be searched by a group of users and proves security by learning with errors (LWE) problem. The traditional privacy-preserving image retrieval scheme not only brings huge computing and communication overhead but also cannot protect image and query privacy well in multiuser scenarios. Wang et al. [27] proposed a content-based privacy-preserving image retrieval (CBIR) framework, which protects the image and query privacy in multiuser scenarios to an extent, and designed a key conversion protocol to support unshared key multiuser image retrieval without loss of search accuracy. To solve the problem of speech search encryption in the cloud, Li and Zhang. [2] proposed a cloud storage method that supports speech encryption search, users use symmetric cryptography algorithm to encrypt local data, and the speech recognizer and hidden Markov model are applied to extract speech text keywords. Glackin et al. [28] presented a strategy for enabling speech recognition to be performed in a cloud environment while preserving the privacy of users. The CNN model is utilized to extract acoustic features of audio and realizes the search of the speech content through the ranking searchable encryption (RSE). Li et al. [29] designed a searchable encrypted voice scheme with privacy-preserving by using a granule computing technique. It uses MFCC to extract voice features and the features obfuscated by creating the fuzzy granule, and AES is adopted to encrypt voices. Then the k-nearest neighbor ciphertext granule search algorithm is used to achieve category retrieval. Hadian et al. [30] proposed a privacy-preserving voice search scheme for medical data, which uses homomorphic encryption to preserve the richness and privacy of voice data and enables efficient voice-based search. Li et al. [31] presented a multiuser searchable encryption voice scheme (MUSEV) deployed in a home IoT system. The LSTM method is employed to achieve voice conversion and enhance retrieval performance and uses the Diffie–Hellman algorithm to exchange parameters between users to support multiuser scheme and improve security.

In summary, most of the existing SE schemes are for multimedia text and image retrieval, and there are relatively few works on the retrieval of sensitive speech data. In response to these issues, an encrypted speech retrieval scheme based on multiuser searchable encryption is proposed. Combined with CP-ABE’s characteristic of data security sharing, the scheme enables multiple users to share and retrieve the encrypted speech data stored in the cloud while protecting the privacy of speech data and keywords and uses the LSTM network model to extract deep semantic features as speech keywords to improve the retrieval accuracy.

3. Preliminaries

3.1. Bilinear Pairing Map. \( G_0 \) is defined as the cyclic group of prime number \( p \), the generator of \( G_0 \) is \( g_1 \) and \( G_T \) is the multiplicative cyclic group with the same order as \( G_0 \). There exists a bilinear mapping \( e: G_0 \times G_0 \rightarrow G_T \) is defined under the following properties:

\[(1) \text{Bilinearity: for all } a, b \in \mathbb{Z}_p, a, b \in \mathbb{Z}_p, g_1, g_2 \in G_0, \text{ there exists } e(g_1^a, g_2^b) = e(g_1, g_2)^{ab}. \]

\[(2) \text{Nondegeneracy: for all } g_1, g_2 \in G_0, e(g_1, g_2) \neq 1_G, \text{ where } 1_G \text{ is the identity element of group } G_2. \]

\[(3) \text{Computability: for all } g_1, g_2 \in G_0, \text{ there exists an efficient algorithm to compute } e(g_1, g_2). \]

\[(4) \text{Symmetry: for all } a, b \in \mathbb{Z}_p, a, b \in \mathbb{Z}_p, g_1, g_2 \in G_0, \text{ there exists } e(g_1^a, g_2^b) = e(g_1^b, g_2^a) = e(g_1, g_2)^{ab}. \]

3.2. Ciphertext-Policy Attribute-Based Encryption (CP-ABE). The attribute encryption is mainly divided into CP-ABE and key policy attribute encryption (KP-ABE). In the CP-ABE scheme, the user’s identity is represented by a set of descriptive attributes, and the user’s key is related to its own set of attributes. The data owner sets the attribute set for the user’s private key and access structure for the ciphertext. Only when the attributes of the users satisfy the access structure, can users decrypt the ciphertext. Otherwise, the ciphertext cannot be decrypted. The CP-ABE scheme mainly includes the following four steps [32]:

Step 1: system initialization: the system setup stage makes the input of the security parameter \( \lambda \), and then it outputs the public parameter \( PP \) and the master key \( MSK \).

Step 2: key generation: the key generation stage takes the public parameter \( PP \) and a set \( I_{\text{CU}} \) of user’s attribute that satisfies the access structure as input; it outputs attribute key \( SK_{I_{\text{CU}}} \), which associates with the attributes of the users.
Step 3: encryption: the encryption stage performs the input of public parameter PP, access structure \( \Lambda \), and a speech data file \( S \) that needs to be encrypted; it produces encrypted data CT of the speech data file \( S \), where CT contains information related to access structure \( \Lambda \).

Step 4: decryption: the encryption stage takes the public parameter PP, encrypted data CT, and attribute key \( \text{SK}_{\text{DU}_i} \) associated with access control attributes list \( L_{\text{DU}_i} \) of users as input. If the attribute set satisfies the access structure \( \Lambda \), then the encrypted data CT can be eventually decrypted by using its attribute private key \( K \).

### 4. The Proposed Scheme

#### 4.1. System Model

Figure 1 shows the system model of the proposed multiuser encrypted speech retrieval scheme with privacy protection. It consists of five different entities: speech data owner (DO), speech retrieval user (DU), trusted authority (TA), cloud server (CS), and proxy server.

The scheme allows multiple users DU, including data owner DO, to retrieve encrypted speech stored in cloud server CS. DO sets the access policy (access control attribute list of users) and uses the CP-ABE scheme to embed the access control policy into the ciphertext. Only the user’s attributes meet the access control policy, and the ciphertext can be decrypted, which achieves a multiuser searchable speech encryption scheme with fine-grained access control.

As shown in Figure 1, the access policy of our scheme adopts the access structure of the “AND” gate that supports multivalued attributes, where “” in the access policy means that the attribute can take any value, realizing flexible access. In our scheme, the access policy is expressed as \( \Gamma = \{L_{00}, L_{11}, \ldots, L_{ij}\} | i = 0, 1, \ldots, m; j = 0, 1, \ldots, n\} \), where \( i \) and \( j \) represent the number of users and user’s attributes, respectively. Assuming that the attribute list of user DU in the system is \( L_{\text{DU}_i} = \{L_{00}, L_{11}, \ldots, L_{ij}\} | j = 0, 1, \ldots, n\} \) and the attribute list of user DU is \( L_{\text{DU}_j} = \{L_{00}, L_{11}, \ldots, L_{ij}\} | j = 0, 1, \ldots, n\} \), then it can be seen that the attribute list of user DU satisfies the access policy, but DU does not meet the access policy, who cannot access and decrypt data. Besides, the proxy server is introduced to perform a partial decrypting operation to reduce computing overhead on the client. The five constituent entities in the system model, as shown in Figure 1, have the following responsibilities:

- **Speech data owner (DO):** DO encrypts the shared speech and stores it to the cloud server (CS). Moreover, multiuser and multiattribute access control lists (access policy) are established by DO. Before encrypting speech data, the LSTM network model is employed to extract deep semantic features of original speech as speech keywords and generates security index I. Then the encrypted speech and security index are bound together and uploaded to the cloud server (CS).

- **Speech retrieval user (DU):** DU generates search trapdoors by extracting keywords of the query speech and sends them to CS for matching retrieval in the ciphertext domain, while users have their own attributes and private key and embed the attributes into the private key to generate attribute private key that is used to decrypt the encrypted speech returned.

- **Trusted authority (TA):** TA is a trusted third-party authority, which is mainly responsible for generating public parameter PP, MSK, and attribute key \( \text{SK}_{\text{DU}_i} \) of the system.

- **Cloud server (CS):** CS provides users with data storage service and matching retrieval operation. After receiving the user’s search request, the search trapdoor uploaded by the user and the security index I stored in the CS are used to calculate the privacy-preserving Euclidean distance to realize speech retrieval in the ciphertext domain. If the search trapdoor is matched successfully, the encrypted speech associated with it will be returned.

- **Proxy server:** the proxy server mainly performs partial decryption operations of the encrypted data to reduce the user’s computing and storage overhead, which is honest and trustworthy and will not disclose any information.

#### 4.2. Security Model

The scheme proposed in this paper selects the indistinguishable security model under plaintext attack [18], which is depicted by introducing the game between challenger B and adversary A, and the specific process is as follows:

- **Initialization:** adversary A sends the access policy \( \Lambda \) to challenger B that he desires to challenge.

- **Setup:** challenger B executes the key generation algorithm and sends the public parameter PP and the master key MSK to adversary A.

- **Phase 1:** adversary A queries the attribute key \( \Lambda \) of the users, but the attribute list \( L_{\text{DU}_i} \) of the user does not satisfy the \( \Lambda \), and then the corresponding attribute key \( \text{SK}_{\text{DU}_i} \) is returned by challenger B.

- **Challenge:** adversary A submits two equal length original data \( S_0 \) and \( S_1 \), then randomly selects \( \beta \in \{0, 1\} \), encrypts \( S_\beta \) under the access policy \( \Lambda \), generating ciphertext \( CT_\beta \), and finally sends it to adversary A.

- **Phase 2:** adversary A repeats phase 1, but adversary A continues to submit a series of attribute list \( L_{\text{DU}_i} \) to challenger B that is different from phase 1.

- **Guess:** adversary A outputs guess \( \beta' \). If \( \beta' = \beta \), adversary A wins the game; otherwise, adversary A loses. The advantage of adversary A in this game is defined as

\[
\text{Adv}_{\text{CPA}}(\lambda) = \frac{|P, [\beta = \beta'] - 1|}{2}.
\]

**Definition:** if A does not break the above security model in a nonnegligible advantage in polynomial time, then our scheme proposed is secure based on the ciphertext indistinguishable under select plaintext attack (IND-CPA).
4.3. Speech Deep Semantic Feature Extraction. In view of the time series relationship of speech features, the LSTM network model is utilized to extract the deep semantic features of the input speech and takes the feature as speech keyword $W$ in this paper. The internal structure of LSTM is shown in Figure 2, which is composed of forget gate, input gate, and output gate.

Compared with the recurrent neural network (RNN), a state vector $C_t$ is added, and forgetting and updating information is controlled by gating unit. The LSTM model proposed in this paper mainly consists of input layer, hidden layer, and output layer, it has 128 structural units, as shown in Figure 2, in the hidden layer, and the hidden layer at the current moment is determined by the previous hidden layer and input at the current moment.

(1) Forget gate: it is used to save partially forgotten information in the original LSTM unit, and according to output $H(t−1)$ of the hidden layer at the previous moment and input $X(t)$ at the current moment, the forgotten degree $f_t$ is output by activation function $\sigma$ (sigmoid function):

$$f_t = \sigma(W_{xf}^T \cdot X(t) + W_{hf}^T \cdot H(t−1) + b_f).$$

(2) Input gate: it decides what new data can be added and stored to the LSTM unit. The input gate $i_t$ is firstly calculated, and a new candidate memory unit vector $\tilde{C}_t$ is generated through the tanh function, which is then added, and the data from the memory unit after forget gate $f_t$ as output $C_t$ of the memory unit:

$$i_t = \sigma(W_{xi}^T \cdot X(t) + W_{hi}^T \cdot H(t−1) + b_i),$$

$$\tilde{C}_t = \tanh(W_{xe}^T \cdot X(t) + W_{he}^T \cdot H(t−1) + b_e),$$

$$C_t = i_t \odot \tilde{C}_t + f_t \odot C_{t−1}.$$ (5)

(3) The LSTM unit of the hidden layer outputs data $y(t)$ by multiplying the data $o_t$ of the output gate and data $C_t$ of the new memory unit processed by tanh function point by point:

$$o_t = \sigma(W_{xo}^T \cdot X(t) + W_{ho}^T \cdot H(t−1) + b_o),$$

$$y(t) = H(t) = o_t \odot \tanh(C_t).$$

where $W_x$, $W_o$, and $W_o$ are parameter matrices to be trained and $b_f$, $b_i$, and $b_e$ are bias items to be trained.

(4) Finally, the data $y(t)$ of the hidden layer is input into the fully connected dense layer containing 512 neurons, and tanh is used as the activation function. The depth feature vector $V = \{V(i)|i = 1, 2, \ldots, l\}$ is obtained in the dense layer, where $l$ represents the dimension, and it is 512 in this paper. Eventually, 512 feature data are mapped into 10 categories of speech in the output layer. To reduce complexity, the extracted deep semantic feature $V$ is hashed according to (7) to construct speech hash sequence $V_s = \{v_1, v_2, \ldots, v_l\}$ as the speech keyword.
4.4.1. Building of Encrypted Speech Library and Secure Index

(1) Key generation is as follows.

Step 1: TA first generates public parameters $PP$ and master key $MSK$ of the system. $g$ is a generator of the multiplicative cyclic group $G_0$, and $P$ is the large prime number. It chooses a pseudorandom function $f: [0, 1]^λ \rightarrow G_0$ mapping to prime number field $Z_p$ that the order is $P$, selects the random numbers $y_j$, $t_0$, $t_1$, ..., $t_n \in Z_p^∗$, and computes $F = e(g, g)^{y_j}, T_i = g^{t_i}$. (8)

The specific steps are as follows.

4.4. Detailed Construction of Multiuser Speech Retrieval Scheme. The system model in Figure 1 studies the searchable speech encryption scheme with multiple users $DU = \{DU_i | i = 0, 1, 2, \ldots, m\}$, where $DU_0$ is the data owner and $m$ is the number of users. The specific processing procedure for multiuser searchable speech encryption is as follows.

Firstly, DO defines an access policy $Λ$ based on the actual situation and encrypts data under this policy $Λ$. The access structure is represented by the access control attribute list of the users so that only users satisfied with the access policy can access and decrypt. The attribute list of the users is exhibited in Table 1.

Table 1: The access control attribute list of the users.

<table>
<thead>
<tr>
<th>User</th>
<th>Attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>DU_0</td>
<td>L_{01} L_{02} ... L_{0n}</td>
</tr>
<tr>
<td>DU_1</td>
<td>L_{11} L_{12} ... L_{1n}</td>
</tr>
<tr>
<td>DU_2</td>
<td>L_{21} L_{22} ... L_{2n}</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>DU_m</td>
<td>L_{m1} L_{m2} ... L_{mn}</td>
</tr>
</tbody>
</table>

Let the public parameter $PP = \{(g, f, F, T_i) | i = 0, 1, \ldots, m; j = 0, 1, \ldots, n\}$ and the master key $MSK = \{y_j, t_i | i = 0, 1, \ldots, m; j = 0, 1, \ldots, n\}$. Step 2: TA uses pseudorandom function $f$ to produce random numbers $R_0, R_0, R_1 \in Z_p$ and random matrix $M \in Z_p^{l \times n}$ with size $(l + 3) \times (l + 3)$ and takes $\{R_0, R_1, M^{−1}\}$ as the key for the construction of security index I of the data owner, and $\{Ra, M\}$ is used as the key to generating search trapdoor $T_q$ of the query user.

(2) According to the access control attribute list $L_{DU_i} = \{L_{ij} | j = 0, 1, \ldots, n\}$ of the users, TA generates the attribute key $SK_{DU_i} = \{sk_{ij} | j = 0, 1, \ldots, n\}$ of the users, where $n$ is the number of users’ attributes.

Step 1: TA selects a random number $r \in Z_p^∗$ and calculates

$$r_{ij} = \frac{rL_{ij}}{\sum_{j=1}^{n} L_{ij}}.$$ (9)

Step 2: TA calculates attribute key $sk_{ij}$ based on the attribute $L_{ij}$ in the attribute set $L_{DU_i}$ of the user $DU_i$:

$$sk_{ij} = \prod_{j=1}^{n} r_{ij} = T_i^{\sum_{j=1}^{n} r_{ij}} = T_i^r.$$ (10)

Then the attribute key $SK_{DU_i}$ of the user $DU_i$ can be expressed as

$$SK_{DU_i} = \{sk_{ij} | sk_{ij} = T_i^r; j = 1, 2, \ldots, n\}.$$ (11)

Step 3: TA computes the private key of the users:

$$b = g^{r_i}, d = g^{rr_i},$$ (12)

where $\{i = 0, 1, \ldots, m; j = 0, 1, \ldots, n\}$. Finally, TA sends the attribute key $SK_{DU_i}$ obtained along with $\{b, d\}$ to the user $DU_i$.

(3) TA encrypts the speech data file $S$ of the user $DU_i$ to generate an encrypted data $CT$.

TA chooses a random number $s \in Z_p^∗$ and $F$ in the public parameter $PP$ and computes

$$C_0 = g^s, C_i = g^s \left( \prod_{j=1}^{n} T_j \right)^s.$$ (13)

Let the encrypted data $CT = \{C, C_0, C_i\}$, and upload $CT$ to CS for storage.
(4) DO generates the security index table corresponding to the speech file $S = \{S_1, S_2, \ldots, S_l\}$.

Step 1: DO first extracts $l$-dimensional deep semantic feature vectors $V = \{V_1, V_2, \ldots, V_n\}$ of the speech file $S = \{S_1, S_2, \ldots, S_l\}$ by using the approach mentioned in Section 4.3.

Step 2: DO selects the random numbers $R_i$ and $R_j$ generated and expands speech feature vector $V$ into $l+3$-dimensional vector $v^l_j$, and the calculation is shown in

$$v^l_j = \left( \sum_{j=1}^{l} v^2_{ij} + R_i - R_j, v_{i1}, v_{i2}, \ldots, v_{il}, 1, R_j \right). \quad (14)$$

Step 3: DO uses the inverse matrix $M^{-1}$ of the random reversible matrix $M$ generated in Step 2 to encrypt the vector $v^l_j$, and the calculation is shown in

$$EM^{-1}(V) = v^l_j M^{-1}. \quad (15)$$

After completing encryption, DO uses $E_k(V)$ as the secure index of the feature vector $V$ corresponding to the speech file $S$ and uploads the security index $I$ to the CS for storage.

4.4.2. User Speech Retrieval

(1) For each query speech $Q = \{q_1, q_2, \ldots, q_n\}$, DU generates search trapdoor $T_q$.

Step 1: DU first uses the method in Section 4.3 to obtain the $l$ dimension deep semantic feature vector $q$ of the query speech $Q$.

Step 2: DU adopts the random number $Ra$ generated to extend the feature vector $q$ to $(l+3)$ dimensions vector $q^l$ by

$$q^l = (1, -2q_1, -2q_2, \ldots, -2q_l, Ra, 1). \quad (16)$$

Step 3: according to the random reversible matrix $M$ generated in Step 2, DU encrypts vector $q^l$ generated that is calculated by

$$EM = (q^l)^T M^{-1}. \quad (17)$$

Finally, the query user DU takes $E_k(q)$ as the search trapdoor $T_q$ of the query speech $Q$ corresponding to the feature vector $q$ and sends it to the CS.

(2) CS performs speech retrieval.

When CS receives the retrieval trapdoor $T_q$ uploaded by the query user DU, the similarity between it and the security index table is calculated by

$$D = I \cdot T_q = E_k(v) \cdot E_k(q) = (v^T M^{-1}) \cdot (q^T M^T)^T$$

$$= \left( \sum_{j=1}^{l} v^2_{ij} + R_i - R_j, v_{i1}, v_{i2}, \ldots, v_{il}, 1, R_j \right)$$

$$\cdot (1, -2q_1, \ldots, -2q_l, Ra, 1)^T$$

$$= \sum_{j=1}^{l} (v_{ij} - q_j)^2 + \left( \sum_{j=1}^{l} q^2_j - R_i - Ra \right)$$

$$= v \cdot q^T + \left( \sum_{j=1}^{l} q^2_j - R_i - Ra \right)$$

$$= dis + \left( \sum_{j=1}^{l} q^2_j - R_i - Ra \right), \quad (18)$$

where $dis$ represents the similarity between query speech feature vector $q$ and feature vector $V$ in the speech library by calculating Euclidean distance, as shown in (19), and $\left( \sum_{j=1}^{l} q^2_j - R_i - Ra \right)$ is a constant, which will not affect the final result. Therefore, it is shown that the similarity measure between
encrypted features is equivalent to plaintext, and the
distance of the ciphertext domain is proportional to
the plaintext domain from (18). Thus, the speech
retrieval in the ciphertext domain is equivalent to
that in the plaintext domain, and the scheme realizes
privacy-preserving speech retrieval in the cloud
environment.

\[
dis = (q, V) = \sum_{i=1}^{l} (v_{ij} - q_j)^2, \tag{19}
\]

where \(l\) is the dimension of the speech feature vector
and \(v_{ij}\) and \(q_j\) represent the eigenvalue of the speech
feature vector in the speech library and the query
speech feature vector, respectively.

(3) Proxy server performs partial decryption of the ci-
phertext CT to obtain a decrypted ciphertext \(E\).

Step 1: proxy server uses private keys \(sk_k\) and \(C_0\) to
partially decrypt ciphertext \(C\) to get the decrypted
ciphertext \(E\).

\[
E = \frac{C}{e(sk_k, C_0)}
\]

\[
= d \cdot sk_k = g^{y_{rij}} \prod_{j=1}^{l} T_i^{y_j}.
\tag{20}
\]

\[
\sum_{j=1}^{n} y_j = y,
\]

\[
\sum_{j=1}^{n} r_{ij} = \frac{r \sum_{j=1}^{n} L_{ij}}{\sum_{j=1}^{n} r_{ij}} = r,
\tag{23}
\]

\[
K = \frac{1}{e(b, C_1)} = \frac{1}{e(g^{y_i}, g^{(\prod_{j=1}^{l} T_i^{y_j})})} = \frac{1}{e(g^{y_i}, g^r) \cdot e(g^{y_i}, (\prod_{j=1}^{l} T_i^{y_j}))}
\]

Proof. \(E/K = C/e(sk_k, C_0)K = C \cdot e(g^{y_i}, g^r) \cdot e(g^{y_i}, \prod_{j=1}^{l} T_i^{y_j}) = C \cdot e(g^{y_i}, g^r) \cdot e(\prod_{j=1}^{l} g^{y_j}, g^r) = SF^x \cdot e(g^{y_i}, g^r) \cdot e(\prod_{j=1}^{l} g^{y_j}, g^r) = SF^x \cdot e(g^{y_i}, g^r) \cdot e(\prod_{j=1}^{l} g^{y_j}, g^r) = S \cdot (\prod_{j=1}^{l} e(g, g)^{y_j}) \cdot e(\prod_{j=1}^{l} g^{y_j}, g^r) = S \cdot (\prod_{j=1}^{l} e(g, g)^{y_j}) \cdot e(\prod_{j=1}^{l} g^{y_j}, g^r) = S \cdot e(g, g)^{\sum_{j=1}^{l} y_j} / e(g, g)^{y_j} = S \cdot e(g, g)^{y_j}
\]

5. Experimental Results and Performance Analysis

This paper evaluates the performance of the proposed
scheme by the theoretical analysis and experimental re-
results. The Chinese speech database THCHS-30 [33]
published by the Speech and Language Technology
Center of Tsinghua University (CSLT) is adopted as the
dataset, with a total duration of more than 30 hours, a
sampling frequency of 16 kHz, and a sampling accuracy of
16-bit single-channel wave format speech segments. Ten
kinds of speech sound with different content were ob-
tained from 17 different speakers, and a speech segment of
equal length 4s was used for the experiment, 80% of
which was used for LSTM training and 20% for testing. In
the experimental retrieval analysis stage, 1000 speech
segments from THCHS-30 were randomly selected for
evaluation.

The hardware platform is Intel(R) Core(TM) i5-5200U
CPU, 2.50 GHz, and 16 GB of memory, and the software
environment is Windows 10, MATLAB R2017b, and Ten-
sorFlow-CPU 2.1.2+Python 3.7 for experimental testing and
performance evaluation.
5.1. Security Analysis. In this section, based on the DBDH assumption, the security against selected plaintext attacks under the standard model is proved, and an analysis of data and search privacy is conducted.

Theorem: challenger B can resolve the DBDH difficulty problem with a nonnegligible advantage $\varepsilon/2$, if adversary A can break the proposed scheme with an advantage $\varepsilon = A d_{\text{CPA}}(\lambda)$.

Proof: challenger B inputs a random DBDH challenge $(g, g^a, g^b, H)$ and a random bit $\sigma \in \{0, 1\}$; if $\sigma = 0$, set $H = e(g, g)^{\alpha\beta}$. Otherwise, if $\sigma = 1$, then $H$ is a random value in $G_T$. The interaction between challenger B and adversary A is simulated as follows.

Initialization: adversary A chooses the access structure $\Lambda$ to be challenged that is sent to challenger B and adversary A. Assumes that the trusted authority TA and the proxy server are loyal and will not disclose the attribute key $SK_{DU}$.

Setup: challenger B sends the public parameter $PP$ and the master key $MSK$ to adversary A.

Phase 1: adversary A submits an attribute list $L_{qij} \in L$ to challenger B requesting an attribute key $SK_{DU}$.

Challenge: adversary A submits two equal length original plaintext data $S_0$ and $S_1$ and randomly selects $\beta \in \{0, 1\}$ to generate challenged ciphertext $C^* = S_0 \cdot H^{\alpha\beta} \cdot \prod_{i=1}^{r} e(g^\theta, g)^{T^i_{\beta}}$, $C_0 = g^\theta$, and $C_0^* = (g^\theta)^{\alpha}\sum_{i=1}^{r} t_i$. Finally, challenger B returns the challenged ciphertext $CT^*_0 = (\Lambda^*, C^*, C_0^*, C_1^*)$ to adversary A.

When $H = e(g, g)^{\alpha\beta}$, the ciphertext $CT^*_0 = (\Lambda^*, C^*, C_0^*, C_1^*)$ is the legal ciphertext of the plaintext $S_0$. Otherwise, when $H$ is a random value in $G_T$, the ciphertext $CT^*_0$ is a random ciphertext.

Phase 2: perform the same query as in the key query phase 1.

Guess: adversary A guesses a bit $\beta' \in \{0, 1\}$; if $\beta' = \beta$, challenger B outputs 0, indicating a guess $H = e(g, g)^{\alpha\beta}$, and then $\sigma = 0$. At this time, $CT^*_0$ is an available ciphertext, and the advantage of adversary A is $\varepsilon = A_{\text{CPA}}(\lambda)$. Therefore, it can be concluded that

$$P_r\left[ B(g, g^a, g^\theta, g^\theta, e(g, g)^{\alpha\beta}) = 0 \right] = \frac{1}{2} + A_{\text{CPA}}(\lambda).$$  

(24)

Otherwise, it outputs 1, which means that the guess $H$ is a random value in $G_T$; that is, $\sigma = 1$. At this point, it is completely random for adversary A, and it can acquire

$$P_r\left[ B(g, g^a, g^\theta, g^\theta, h, e(g, g)^{\alpha\beta}) = 0 \right] = \frac{1}{2}$$  

Thus, the advantage of challenger B is

$$P_r\left[ B(g, g^a, g^\theta, g^\theta, e(g, g)^{\alpha\beta}) = 0 \right] = \frac{1}{2} + A_{\text{CPA}}(\lambda) = \frac{\varepsilon}{2}.$$

(25)

Therefore, it can be concluded that the DBDH assumption can be broken with a nonnegligible advantage. The scheme proposed is IND-CPA secure.

(1) The privacy protection of the speech data

In our scheme, according to $C = SF^a$ in (13), using public parameter $F$ encrypts the original data $S$ to obtain the encrypted data $C$, and only users satisfied with the access policy $\Lambda$ can decrypt the ciphertext $C$, effectively preventing illegal users from accessing and acquiring data and improving the security of the data. Even if the malicious user acquires the ciphertext $C$, according to the attribute key generation (10), the adversary A first needs to solve the discrete logarithm equation $sk_{T_0} = T_0^{\beta}$ in (11). Due to the difficulty in calculating the discrete logarithm problem, $r_q$ cannot be solved.

(2) The privacy protection of the keywords

According to (15), the keywords corresponding to each speech file are encrypted with the randomly reversible matrix $M^{-1}$ and the extended random vector $\vec{v}_i$ to hide its content. The encryption result is affected by the reversible matrix $M^{-1}$ and random numbers $R_i$ and $R_q$, and these random numbers are indistinguishable. Therefore, the same keyword generates different ciphertext under the same key, and adversary A cannot obtain any information about plaintext from the ciphertext in the polynomial time, which ensures the safety of the keywords.

(3) The privacy protection of the search trapdoor

In the process of generating search trapdoor $T_q$, according to (17), the query user firstly selects random number $Ra$ for each query feature $q$ to construct a random vector $q^\theta$ and then uses a random matrix $M$ to encrypt the vector to generate search trapdoor $T_q$. Since $Ra$ and $M$ are indistinguishable, different search trapdoor $T_q$ can be generated for the same query. If adversary A obtains the trapdoor, it cannot get indistinguishable random number $Ra$ and random matrix $M$ from the trapdoor $T_q$, thus, no information about $Q$ can be acquired that protects the search privacy.
5.2. Performance Analysis of the Proposed LSTM Network Model. At present, the method of employing the LSTM network model enabling speech recognition can be used as a keyword extraction approach to solving the issue of speech keyword extraction. Therefore, the proposed scheme uses LSTM deep learning model to extract deep semantic features of speech and evaluates the performance of the model through the pretraining model. Figure 4 shows the training/test curve of the LSTM network model under the speech spectrogram and MFCC features.

As shown in Figure 4, taking MFCC as input of the LSTM network model has faster convergence, lower loss, and higher accuracy. After 40 iterations of training, its training accuracy is 99.97%, and the test accuracy is 99.28%; meantime, the training and test loss is almost close to 0, and the model is not overfitting. However, when the speech spectrogram is used as the input, its training performance is poor. This is because the speech spectrogram feature contains too much redundant information and has a large dimension, which will cause the loss of the feature information of the spectrogram image when it goes through the LSTM network model.

5.3. Efficiency Analysis. Table 2 shows the computational complexity analysis of the proposed scheme, where \( n \) represents the number of speech files, \( l \) is the dimension of speech feature vectors, and \( \text{DOT}_{(l+3)} \) represents the inner product operation of two \((l+3)\) vectors. In the stage of constructing the security index table, according to (14), each speech file needs to perform \((l+3)\) inner product operation of feature vector and reversible matrix, so \( n(l+3) \) inner product operation should be performed, and the computational complexity is \( n(l+3) \) \( \text{DOT}_{(l+3)} \). Similarly, in the process of the generation of the search trapdoor \( T_v \) due to the fact that the scheme is full-text matching retrieval, its computational complexity is \( n(l+3) \) \( \text{DOT}_{(l+3)} \) according to (16). When performing speech retrieval, CS needs to execute \(|W|\) times vector inner product operation, and its computational complexity is \( n^2(l+3) \) \( \text{DOT}_{(l+3)} \). For the storage overhead, the calculation results are shown in Table 3, where \(|Z_p|\), \(|G_v|\), and \(|G_T|\) represent the length of elements, and \(|I_q|\) is the number of attributes in the scheme.

Figure 5 shows the performance comparison of encryption time using attribute encryption for original speech data with a different number of attributes and the size of the speech data.

As shown in Figure 5(a), the time of key generation increases with the increase of the number of user attributes, while the data encryption and decryption time remain basically unchanged or the decryption time changes in a relatively stable range. It can be seen from Figure 5(b) that as the size of the speech data increases, the time of encryption and decryption is also increasing. Therefore, the scheme proposed in this paper transfers the key generation task to a trusted authority TA; meantime, part of the decryption operation is outsourced to proxy servers, which can effectively reduce the computing time of the users and improve the efficiency of the system.

5.4. Evaluation of Retrieval Performance. The recall ratio (R), precision ratio (P), F1 score, and retrieval time are usually used to measure the performance of speech retrieval algorithms. The calculation methods of recall rate R and precision rate P are shown in (27) and (28), respectively.

\[
R = \frac{TP}{TP + FN} \times 100\%, \tag{27}
\]

\[
P = \frac{TP}{TP + FP} \times 100\%, \tag{28}
\]

where \( TP \) is the number of samples in the retrieval results that are related to the query keywords and judged to be positive; \( FN \) represents the number of samples that are related to the query keywords but judged to be negative; \( FP \) is the number of samples that are not related to the query keywords but judged to be positive.

In order to further test the performance of speech retrieval, the F1 score is usually used to evaluate the system performance. The higher the F1 score is, the better the performance is. It can be calculated by

\[
F1 = \frac{2 \times P \times R}{P + R}. \tag{29}
\]

Existing studies have shown that the precision-recall (P-R) curve can directly and comprehensively reflect the performance of the speech retrieval scheme. Figure 6 shows the comparison results of the P-R curve for the proposed scheme under two different features.

As shown in Figure 6(a), the AUC values obtained from the ROC curves with the two different features are 0.81 and 0.89, respectively. The value range is all from 0.5 < AUC < 1, indicating that the proposed LSTM model has good generalization ability, and when MFCC is used as input, the classification effect is more significant. Besides, the area enclosed by the P-R curve and X-Y coordinate axis is larger when the input is MFCC features, as shown in Figure 6(b), so it acquires better retrieval performance. Moreover, because recall R and precision P are mutually affected, the proposed scheme has the greatest influence on precision P when recall R is 1.

In the experiment, the paper compares the retrieval performance of the proposed scheme with MFCC and spectral acoustic features as input of the LSTM network model, as shown in Figure 6. In the proposed scheme, 20 Mel cepstral coefficients are extracted from each speech frame, the speech spectrogram is produced for each speech frame by using the STFT feature, and the spectrogram image is scaled to 227 × 227 size to reduce the amount of data calculation. During the retrieval stage, 1000 speech fragments from the THCHS-30 database were randomly selected as the speech to be retrieved by authorized users after the operation of MP3 compressed content retention, and the deep semantic features of the speech to be retrieved were extracted as search keywords to execute retrieval operation by using the method in Section 4.2. When performing retrieving, the similarity threshold \( T \) is set. It shows that the retrieval is successful if the Euclidean distance \( \text{dis}(q, V) < T \) between a query keyword and a keyword stored in the secure index I in
Figure 4: The speech spectrogram and MFCC train/test curve. (a) The train/test loss curve. (b) The train/test accuracy curve.

Table 2: The computational complexity analysis of the proposed scheme.

<table>
<thead>
<tr>
<th>Secure index building</th>
<th>Search trapdoor ( T_q ) generation</th>
<th>Speech retrieval</th>
</tr>
</thead>
<tbody>
<tr>
<td>( n(l + 3) ) DOT ( (l,e) )</td>
<td>( n(l + 3) ) DOT ( (l,e) )</td>
<td>( n^2(l + 3) ) DOT ( (l,e) )</td>
</tr>
</tbody>
</table>

Table 3: The storage and communication overhead.

<table>
<thead>
<tr>
<th>Secure index</th>
<th>Search trapdoor</th>
<th>Public parameter</th>
<th>Attribute key</th>
<th>Ciphertext ( CT )</th>
<th>Communication</th>
</tr>
</thead>
<tbody>
<tr>
<td>( 2n(l + 3)| Z_p | )</td>
<td>( 2n(l + 3)| Z_p | )</td>
<td>( 6| G_0 | + | G_T | )</td>
<td>( (2| L_\alpha | + 3)| G_0 | + | Z_p | )</td>
<td>( | G_T | + | 3l + 2| G_0 | )</td>
<td>( 2| Z_p | )</td>
</tr>
</tbody>
</table>

Figure 5: Performance comparison of the encryption time. (a) Encryption time under a different number of attributes. (b) Encryption time in different speech data size.
the cloud, and the similarity threshold $T$ is set as 0.245 in this paper. In order to test the retrieval performance of the proposed scheme, the performance is compared with that of the method in [31,34]. Table 4 shows the comparison results of retrieval performance between the proposed scheme and the existing schemes.

As shown in Table 4, the proposed scheme has obtained a higher accuracy, recall rate, and F1 score, indicating that the proposed scheme has good retrieval performance. However, the retrieval efficiency of the proposed scheme is about twice that of [31,34]. However, when MFCC is used as the input of LSTM, the retrieval accuracy, recall rate, and F1 score all reach 100%; this is because the hash construction was performed, effectively enhancing the retrieval performance. When taking the spectrogram images as input, F1 scores were about 10% lower than those of the two schemes in [34], but the retrieval recall rate is equivalent. Compared with [31], the retrieval accuracy and recall rate of scheme-A are lower than those in [31]; however, compared to scheme-B, the recall rate and F1 score are similar. Therefore, the LSTM network is used as the method of the speech content keyword extraction in this paper with better retrieval performance.

### 6. Conclusions

This paper proposes an encrypted speech retrieval scheme based on multiuser searchable encryption in cloud storage, which not only can solve the issue of privacy disclosure and secure speech data sharing for multiple users in the process of the encrypted speech retrieval in cloud storage but also realizes the efficient and secure retrieval of massive encrypted speech and employs LSTM network model to extract deep semantic features that further improve the retrieval accuracy and recall rate. In order to enhance the security of speech data and achieve access control between users, the CP-ABE scheme is adopted to protect the privacy of the data and realize fine-grained access control of multiple users. A privacy-preserving encrypted speech retrieval is realized by encrypting speech features with a random matrix and matching retrieval with Euclidian distance after hiding keywords content. The security analysis and experimental results indicate that the proposed scheme can satisfy the indistinguishability of the index and trapdoor information, has higher retrieval performance, and also proves that the scheme chooses security under the chosen plaintext attack based on the DBDH assumption.

<table>
<thead>
<tr>
<th>Scheme</th>
<th>Multiuser</th>
<th>Security</th>
<th>P (%)</th>
<th>R (%)</th>
<th>F1 (%)</th>
<th>Retrieval time (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ref. [31]</td>
<td>Scheme-A</td>
<td>Diffie-Hellman</td>
<td>94.00</td>
<td>93.00</td>
<td>93.50</td>
<td>0.139</td>
</tr>
<tr>
<td></td>
<td>Scheme-B</td>
<td>Adavtively</td>
<td>83.00</td>
<td>84.00</td>
<td>83.50</td>
<td>0.134</td>
</tr>
<tr>
<td>Ref. [34]</td>
<td>Scheme-I</td>
<td>CPA</td>
<td>92.00</td>
<td>93.00</td>
<td>92.50</td>
<td>0.150</td>
</tr>
<tr>
<td></td>
<td>Scheme-II</td>
<td>CPA</td>
<td>93.00</td>
<td>92.00</td>
<td>92.50</td>
<td>0.120</td>
</tr>
<tr>
<td></td>
<td>MFCC</td>
<td>IND-CPA</td>
<td>100</td>
<td>100</td>
<td>100</td>
<td>0.269</td>
</tr>
<tr>
<td></td>
<td>Spectrogram</td>
<td>CP-ABE</td>
<td>74.97</td>
<td>92.03</td>
<td>82.63</td>
<td>0.274</td>
</tr>
</tbody>
</table>
In future work, we plan to focus on the research of the ciphertext location leakage and efficient encrypted speech retrieval scheme supporting fuzzy keyword query.

Data Availability

Previously reported speech data were used to support this study and are available at https://arxiv.org/abs/1512.01882. This is cited at relevant places within the text, such as reference [33].

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (nos. 61862041 and 61363078).

References


