Scalable Computing: Practice and Experience, ISSN 1895-1767, http://www.scpe.org
© 2024 SCPE. Volume 25, Issues 3, pp. 1799–1806, DOI 10.12694/scpe.v25i3.2711

A MULTI-SENTENCE MUSIC HUMMING RETRIEVAL ALGORITHM BASED ON RELATIVE FEATURES AND DEEP LEARNING

YELIN ZHANG*

Abstract. This project will study a fast retrieval method for music humming speech recognition based on sentence features and deep learning. The method proposed in this paper can realize the fast extraction of songs. According to the characteristics of the natural pause mode of the song, the song database and the song fragments provided by the user are divided into different sentences. The deep learning algorithm of BDTW is used to calculate the similarity of the song's pitch, and users can set matching conditions according to their preferences. It can identify the most significant differences between music fragments and the order of queries in the database. Then, a retrieval method of a music database based on DIS is proposed. It can shorten the acquisition time. Experiments show that the algorithm can recognize humming songs quickly and efficiently.

Key words: Related features; Deep learning; The songs hum; BDTW algorithm; Search algorithm

1. Introduction. Digital music has gradually developed and formed a popular model in recent years. Now, China's online music industry is growing. Total mobile music sales in China reached 5.987 billion yuan in the first three months of this year. The number of mobile music users has reached 836 million. Major Internet sites currently store millions of electronic music. This makes it more complicated for users to search, retrieve, and find relevant songs. Now, the central system in the industry is in the form of a manual query. Such searches are often based on metadata such as artist name, song title, etc. The few semantically expressive features are musical styles. Usually, only the words mentioned above or other written information can be used when searching for music. In addition, the system only provides the most basic music recommendations and personalized service. It does not depend on the content of its musical signal. The system also provides only one user feature document generation method based on user information. In such a system, the user is represented as a vector, a measure of the click-through rate or number of plays. With the help of a vector library, we can realize the joint recommendation of similar users and similar songs. The same can be done by describing documents according to semantically labelled users, finding similar users or finding a music document. Realizing the automatic, accurate and fast location of music objects is an urgent problem in music notation.

Reference [1] describes a method that indirectly uses user-listening behaviour to generate semantic description documents. They took advantage of users' listening habits and metadata extracted from users' private music profiles. Music services like last.fm are available, as are reviews, biographies, journals, and music-related RSS links on the World Wide Web. Literature [2] uses joint filtering to implement music recommendations. It has an excellent statistical effect on popular songs. But there is a big downside to this. For example, the required user click rate, social tags and other metadata are missing for non-hit songs. Some signs using sound-based advice can ameliorate this problem.

At present, object-oriented analysis and fusion technology are rare. But this approach has its drawbacks. Each of them uses tone and rhythm to convey their meaning. This sound information is low-level and cannot be directly translated into high-level semantic information. Some recent experiments prove that the semantic gap can be bridged by corresponding work in semantics. The "semantic gap" is mainly reflected in the lack of correlation between the low-level feature information in speech and the semantic information of the human brain. Much research has been done on the current music content search methods. Literature [3] gives a new kind of music ontology aiming at the problem of the "semantic gap" problem. A new semantic ontology of music is constructed by using the characteristics of the lower and higher levels of music. Literature [4] provides a semantic model based on spatial context. The technology of situational cognition defines the semantic meaning.

*Zhengzhou Technology and Business University, Zhengzhou 450000, China (Corresponding author, 20100001@ztbu.edu.cn)

It constructs a text-oriented music retrieval system that includes emotion and non-emotion. Semantic features are used to analyse the similarity of search results. Reference [5] uses Dirichlet mixed modes to propose a token-based semantic polynomial. Some background information is added when annotating songs to improve the accuracy of annotations.

When labelling music, literature [6] introduces a method to synthesize the characteristics of various sound sources. Two phonological features and two social features are used in this method. Literature [7] proposes an algorithm based on social labelling and compares it with vector space models, singular value matrix factorization, non-negative matrix factorization, and probabilistic latent semantic analysis. Simulation results show that this method has a better effect than other methods. Literature [8] combines fuzzy music scene features and sound features into the ACT algorithm. The accuracy of content-based music retrieval is improved by using the fuzzy music scene characteristic with expressive semantics. Literature [9] uses an algorithm that uses the compressed string as the time characteristic of music to calculate the similarity of music. The semantic description-based query proposed in the literature [10] is a more natural query method. A good music information search model is developed. However, the biggest obstacle of this algorithm is the lack of clearly labelled, public and open, heterogeneous labelled song data sets. A CAL500 database was constructed in reference [11]. The study fuses users' listening habits with song labels to obtain semantic associations between lyrics and music. Multiple guide classifiers are used to label the pattern. By training the CAL500 library, a song search method based on the CAL500 library is obtained. This mode cannot only tag a new track but also perform text queries and corresponding return values for multiple tracks.

Humming query is a significant component of song retrieval. This search method retrieves the nearest k songs from a single lyric. The research of song extraction technology is relatively backward. A pioneering exploration of humming queries was made in literature [12]. The researchers used a rough string alignment algorithm to complete the humming search. Literature [13] adopts Dynamic Time Organization (DTW) technology to perform complete sequence matching between humming songs and other tracks in the database. Literature [14] uses the N-gram backward index structure in the music data mined by the topic, thus improving retrieval efficiency. Literature [15] uses subsequence as a matching method to solve the problem of missing specific notes or syllables in humming fragments. In addition, the rapid development of string-matching technology in recent years has also played an excellent guiding effect for music search. Literature [16] proposes a multi-dimensional music sequence matching technique. His research uses a combination of sentence features and deep learning to identify songs. The method proposed in this paper can realize the fast extraction of songs. According to the characteristics of the natural pause mode of the song, the song database and the humming fragments provided by the user are divided into different song sentences. The deep learning algorithm of BDTW is used to calculate the similarity of the intonation of humming songs, and users can set matching conditions according to their preferences. It limits the maximum difference between music fragments and the order of queries in the database.

2. Humming music retrieval system framework based on sample semantics. In the humming music retrieval system based on sample semantics, the focus of work is no longer to extract the features of intensity, pitch, beat and melody from the audio signal. First, the recognition accuracy cannot be significantly improved because of the "bottleneck" in recognizing sound level. Secondly, there is no clear research on the relationship between the meaning of humming songs and the sound level. Meaning is often determined by more than just one or a few sound features [17]. The traditional way of acquiring speech features based on speech information is faced with the problem of the "semantic gap." The construction of a music search system based on the example semantics is described in Figure 2.1 (image cited in Bioengineering 2023, 10(6), 685). This project intends to use deep learning methods to find the relationship between raw signals and semantics. The music in the established music library is classified and the semantic analogy is made. This search method can get the user's search intention more naturally and accurately.

3. Music retrieval algorithm based on humming. A sentence-based contour feature library of humming music is established. Calculating the front and back notes can determine the piece's profile. Its composition is "sentence length, first note interval, last note interval, the difference between adjacent notes interval."

According to the above characteristics, the tunes of multiple sentences are searched, and the corresponding characteristics are required to match the tunes of the songs. It requires the production of a candidate set of

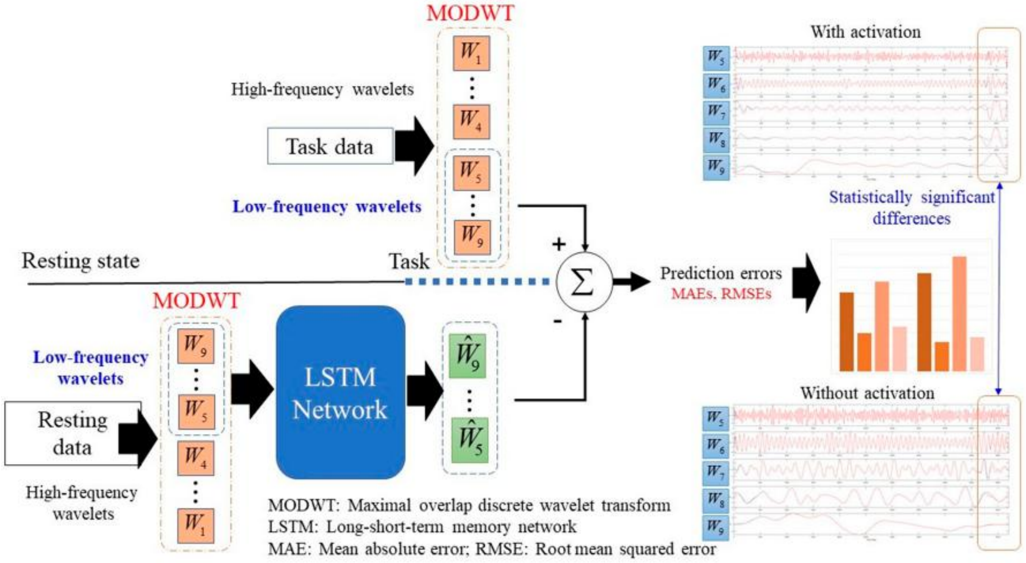

Fig. 2.1: Humming music retrieval model based on sample semantics.

music [18]. By summing the sentence length of adjacent sentences, the corresponding candidate music can be obtained, and the eigenvalue of the candidate music can be extracted. The sentence length feature is the number of music features retained in the music library in sentence units. This is the distance between two consecutive sounds in a sentence. It extracts the number of features in the sentence the user wants to hum. There are two ways to generate eigenvalues:

a) Direct generation of the characteristics of each tonal profile. When the contour features of two sentences are connected, a feature value needs to be added between the two sentences. This value is the interval difference between the latter sentence's first note and the previous sentence's last note. However, when the user connects two sentences, it is easy to make the connection between them inaccurate. Increasing the value of this feature results in lower detection efficiency.

b) Ignore the eigenvalues between different sentences. In this case, it just matches the attributes in the sentence.

3.1. Alternate feature extraction algorithm in multi-sentence search. By preprocessing, feature extraction and melody contour extraction of the song fragment that the user wants to search, the feature string is: $F = \{f_1, f_2, \cdots, f_l\}$. Its sequence is l. Any track in a music library $S = \{L_1, L_2, \cdots, L_d\}$ can be represented by $L_i(1 \leq i \leq d)$. The set of sentences is$L_i = \{R_1, R_2, \cdots, R_{lr}\}$.$R_i(1 \leq i \leq lr$ is for any line in a song. lr is the number of sentences in the entire song. Rearrange the letters that make up R_i into $R_i = \{\varphi_1, \varphi_2, \cdots, \varphi_n\}$. Where n is its length. A multi-sentence search algorithm is used to determine the candidate set of music fragments:

Definition 1. A collection of candidate music pieces. Include the following in the database

$$L_i = \{R_1, R_2, \cdots, R_{lr}\} \tag{3.1}$$

The sentence length is ordered as $D = \{d_1, d_2, \cdots, d_{lr}\}$. The length of the track F to be retrieved is ld. Its fragment G is identified as

$$\begin{aligned} &G = \{R_i + R_{i+1} + \cdots + R_j|| \\ &d_i + d_{i+1} + \cdots + d_j - ld| \\ &\leq \varepsilon_G, (1 \leq i \leq j \leq k)\} \end{aligned} \tag{3.2}$$

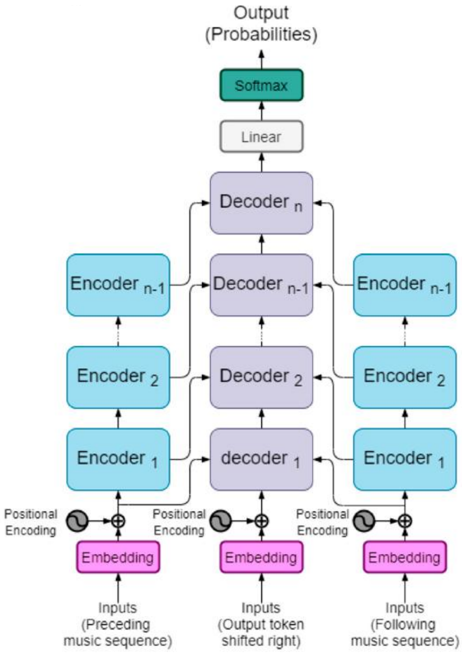

Fig. 3.1: Music candidate segment generation process diagram.

ε_G is the upper limit of the allowed length set by yourself. That's the most significant difference between the two pieces. The following is a more detailed description of the candidate feature generation algorithm for multi-sentence search.

3.1.1. Produce an alternate fragment. The algorithm in Figure 3.1 generates a set of musical candidate pieces G based on the above definition.

3.1.2. Generate alternative feature quantities. The corresponding eigenvalues are generated in the following two ways based on the candidate fragment set:

1) The generation of corresponding tonality profiles, respectively. The method of producing the corresponding melody profile characteristics from the resulting set of alternate music fragments is shown as follows:

(a) The characteristics of the melody profile are generated. When the user hums multiple tunes, the extracted feature quantity is the overall feature quantity composed of multiple tunes [19]. If it is multiple notes, adding one note to the two notes is required. This value is the interval difference between the following sentence's beginning note and the previous sentence's end note. Then the characteristics of two consecutive sentences R_1 and R_2 are:

The first segment of R_1 + the first note interval of R_2.

The interval characteristic of the difference between the end of R_1 and the interval of R_2.

Then, the profile characteristics of the candidate tunes are obtained.

(b) Characteristics of musical rhythm. The tune of the label is characterized by $\Theta(query\ rhythm\ leng th)$.

$$\Theta = \{\theta_1, \theta_2, \cdots, \theta_{lr}\} \tag{3.3}$$

The musical length field is the musical length of each sentence in a song, $\phi(da\,tabase\ \ rhythm\,length)$.

$$\phi = \{\varphi_1, \varphi_2, \cdots, \varphi_n\} \tag{3.4}$$

Let's say $N = \min(msn)$. Let's define $\Delta[N] = \{\frac{\theta_1}{\varphi_1}, \frac{\theta_2}{\varphi_2}, \cdots, \frac{\theta_N}{\varphi_N}\}$. Calculate $dist_{rl} = \sum_{i=2}^{N} |\Delta[i] - \Delta[1]|$. Using the dynamic programming method, the minimum $dist_{rl}$ value can be obtained.

2) Do not consider the specific number of features between sentences. Without considering the eigenvalues between sentences, the eigenvalues are calculated as follows:

For users of multiple speech sentences, the number of features in speech is continuous. These two sentences have an abstract character, but the two coherent sentences make the user hum inaccurate. Multiple features are often extracted from a coherent Chinese character when feature extraction is carried out. When the feature of the armature is set to ”?” It can be matched with any number of characters. In this way, you can get the characteristics of multiple sentences that the user is humming.

3.2. Humming multi-sentence retrieval algorithm. A speech recognition method based on DTW (Dynamic Time Warping) is proposed to solve the problem of pitch inaccuracy caused by speech errors in humming recognition. DTW is a nonlinear rule that combines timing rules with distance measurement rules, and it is widely used in speech recognition. This method is used to compare the embedded-remove errors of a certain class of characters. Thus, the optimal matching subsequence is obtained and its similarity is maximized [20]. If the corresponding features of the two substrings are inconsistent, the similarity definition proposed in this paper is used to analyse the similarity of the two substrings. Here is a description of the algorithm:

1) Extract user humming feature string.

$$R = {'}\varepsilon_1\varepsilon_2 \cdots \varepsilon_n \cdots \varepsilon_N{'}(N \geq 0) \tag{3.5}$$

2) The corresponding feature sequence is obtained by classifying it based on the length N of the sequence removed by humming.

$$T = {'}\sigma_1\sigma_2 \cdots \sigma_l \cdots \sigma_L{'}(L \geq 0), |L - N| < \varepsilon \tag{3.6}$$

3) Search for humming songs according to the DTW algorithm.

The research focus of DTW is to find the time function $l = w(n)$ with certain regularity. The algorithm corresponds a time history n of an input byte to a time history l of a reference byte nonlinearly. And w satisfies$ldis = \min\limits_{w(n)} \sum\limits_{n=1}^{N} d(\sigma_l, \varepsilon_{w(n)})$. Under the best time rule, $d(\sigma_l, \varepsilon_{w(n)})$ is the measure of distance between two fields. The algorithm for approximate string matching based on the stated distance looks like this: Type the string $R = {'}r_1 r_2 \cdots r_n \cdots r_n{'}(n \geq 0)$. Each$R$ here represents a different record in the database.

Output some data that is highly similar to A.

1) The spacing matrix S of corresponding characters in T and R is recorded in the article i of the calculation library.

2) The optimal route is obtained Through dynamic programming of the model.

3) Starting from the definition of similarity, the minimum and the difference between the two lines and their lengths can be obtained to obtain the similarity of T and R. Store a similar value. Add 1 to the value of i and return to a), and so on—to the last record.

4) When all the similarity values are taken as the result of the closest data among the multiple similarity values. In this way, the user can sing any melody, find the target song, and thus achieve multiple sentence searches.

4. Experimental analysis. Two actual databases verify the proposed method. The MIR-QBSH dataset contains 2,048 songs in MIDI format. A total of 4,431 segments were questioned. The singer sings from the beginning of the song. The IOACAS data contains 298 formatted songs and 759 entries from MIDI. The singer starts singing anywhere in a song. The test data was sampled at 8 kHz and stored in 8-bit resolution Wav format. It is converted to an FM sequence by an FM tracking device. The research work in this paper is carried out on Intel’s Q8400 CPU (2.66 GHz) and 2 GB microcomputer. The 64-bit Ubuntu12 system was used. Write programs in C++.

4.1. Verification and analysis of search results based on segmentation of songs. Firstly, the sentence features are extracted from the tonality sequence. Divide the music into different sections according to the sentence. Take the tonality position where tonality 0 occurs consecutively in the tonality sequence as a sentence. The tunes the user hums are usually higher or lower than the original tunes. The pitch the user

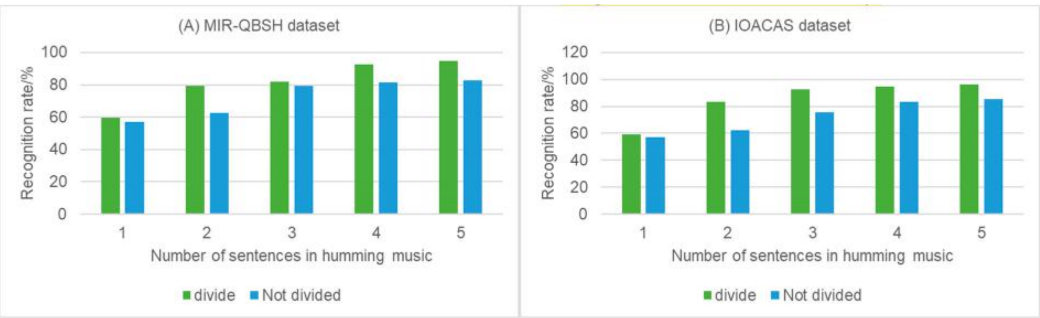

Fig. 4.1: Test effect of music division statement retrieval recognition rate.

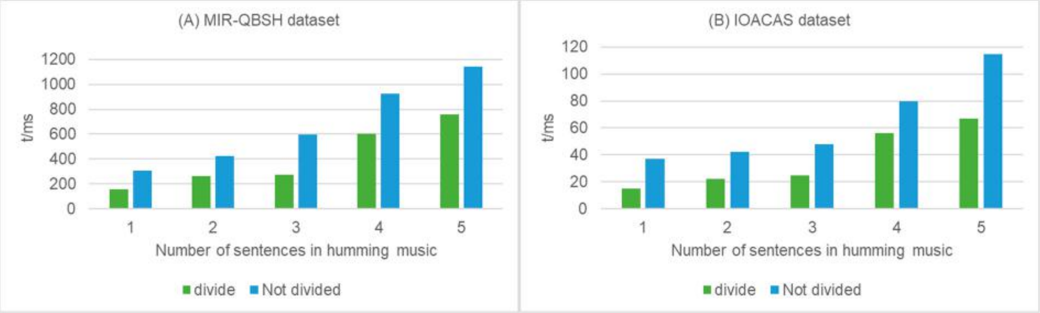

Fig. 4.2: Test effect of music division statement retrieval efficiency.

is humming must be normalized along with the songs in the database. Each point is then modified to be the ratio of its original value to the average height.

We will take MIR-QBSH and IOACAS as research objects to compare BDTW regarding retrieval efficiency and retrieval speed of segmented music sentences. Figuress 4.1 and 4.2 show the identification rate and efficiency of the method. The value of the fault tolerance limit factor should be determined according to the user's feelings about the song. In the experiment, the value of a is 10%. The paper divided the data group into five types based on the number of sounds made by the people in the data group. From Figures 4.1 and 4.2, we can see that the search based on music sentences performs better in all 5 cases regarding efficiency. Especially in the case of a large number of statements, the superiority of this method is more pronounced. This is primarily due to the use of sentences to distinguish, more in line with people's habits of lyrics. The segmentation of the music sentence prevents the mismatching of the front of the unsegmented music sentence from affecting the humming tune. This results in a significant increase in efficiency.

4.2. A method for searching songs by using sentence characteristics. This part of the experiment compares BDTW with similar DTW on a data set. The error constraint factors a is 10%. This paper compares the performance test of the DTW algorithm and DTW algorithm in execution efficiency and execution speed when the return result is top-1, top-5, top-10, top-15 and top-20 under the DIS index structure. The results are shown in Figures 4.3 and 4.4. BDTW search results are better. In DIS, the larger the k value is, the better the retrieval result is. BDTW has better performance than DTW. For top-5 problems, BDTW has obtained a high identification rate. BDTW showed high results in the search of songs, while the total search time spent in the search process remained the same.

5. Conclusion. BDTW algorithm allows users to give fault tolerance limit factors according to their humming level. This algorithm can solve the maximum difference between the restricted database and the query order. This completes a complete sequentially matched musical statement. The exponential structure of

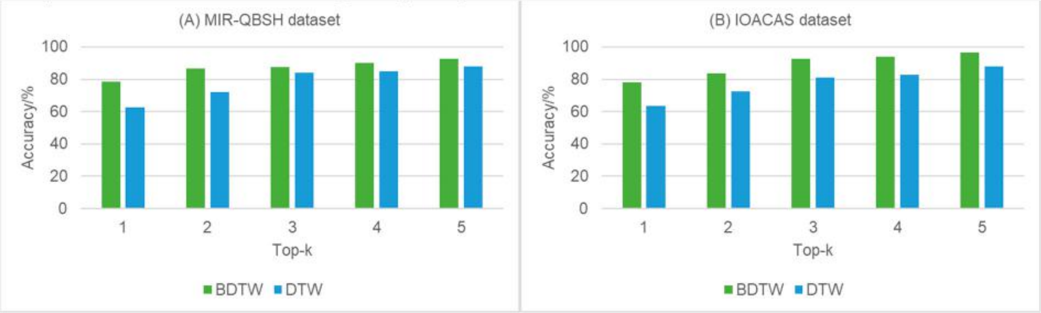

Fig. 4.3: Retrieval accuracy test effect of BDTW algorithm.

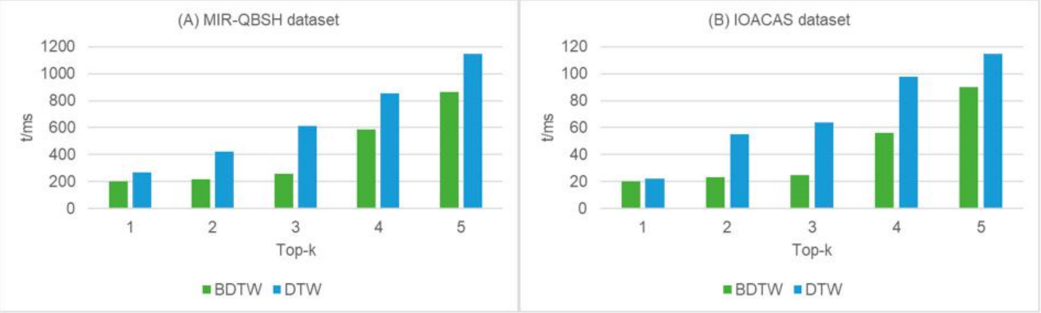

Fig. 4.4: BDTW retrieval efficiency test effect.

DIS is also given. This method is to construct an index structure in the database to reduce the query speed and achieve fast data retrieval. Experimental results show that the algorithm can accurately retrieve the humming problem quickly and efficiently.

REFERENCES

[1] Lee, K. Y., & Hu, C. M. Research on the development of music information retrieval and fuzzy search. Scientific and Social Research, 2022; 4(4):1-10.

[2] Kappen, P. R., Beshay, T., Vincent, A. J., Satoer, D., Dirven, C. M., Jeekel, J., & Klimek, M. The feasibility and added value of mapping music during awake craniotomy: A systematic review. European Journal of Neuroscience, 2022;55(2):388-404.

[3] Muthoifin, M., Ali, A. B. E., Al-Mutawakkil, T., Fadli, N., & Adzim, A. A. Sharia Views on Music and Songs: Perspective Study of Muhammadiyah and Madzhab Four. Demak Universal Journal of Islam and Sharia, 2023;1(01): 10-17.

[4] Carvalho, M. E. S., de Miranda Justo, J. M. R., Sá, C., Gratier, M., & Rodrigues, H. F. Melodic contours of maternal humming to preterm infants in kangaroo care and infants' overlapping vocalizations: A microanalytical study. Psychology of Music, 2022;50(6):1910-1924.

[5] Carvalho, M. E., Justo, J. M., Gratier, M., & Ferreira Rodrigues, H. Infants' overlapping vocalizations during maternal humming: Contributions to the synchronization of preterm dyads. Psychology of Music, 2021;49(6):1654-1670.

[6] Rezaei Oshaghi, N., Baradar, R., & Ghaebi, A. Methods of searching music information sources in search engines. Librarianship and Information Organization Studies, 2023; 33(4): 38-58.

[7] Bakouros, S., Rarey, K., & Evered, J. Retinopathy of Prematurity Screening Exams, Adverse Events, and Music Therapy: A Case Series. Music Therapy Perspectives, 2023; 41(1): 47-53.

[8] Rushton, R., Kossyvaki, L., & Terlektsi, E. Music-based interventions for people with profound and multiple learning disabilities: A systematic review of the literature. Journal of Intellectual Disabilities, 2023; 27(2): 370-387.

[9] Pridy, C. B., Watt, M. C., Romero-Sanchiz, P., Lively, C. J., & Stewart, S. H. Reasons for listening to music vary by listeners' anxiety sensitivity levels. Journal of music therapy, 2021;58(4): 463-492.

[10] Baptista, A., & da Silva, C. G. Organization and Representation of Musical Information (ORMI) in Portugal: a literature review. Boletim do Arquivo da Universidade de Coimbra, 2021; 34(2): 11-26.

[11] Zalkow, F., Brandner, J., & Müller, M. Efficient retrieval of music recordings using graph-based index structures. Signals,

2021; 2(2): 336-352.

[12] Tamboli, A. I., & Kokate, R. D. Query based relevant music genre retrieval using adaptive artificial neural network for multimedia applications. Multimedia Tools and Applications, 2022; 81(22): 31603-31629.

[13] Lee, B. H., & Kim, M. 2021; Algorithm to Search for the Original Song from a Cover Song Using Inflection Points of the Melody Line. KIPS Transactions on Software and Data Engineering, 10(5): 195-200.

[14] Bosher, H. Sheeran succeeds in 'Shape of You'music copyright infringement claim. Journal Of Intellectual Property Law and Practice, 2022;17(7): 544-546.

[15] Fisher, M., & Rafferty, P. Current Issues with Cataloging Printed Music: Challenges Facing Staff and Systems. Cataloging & Classification Quarterly, 2023; 61(1): 91-117.

[16] Velankar, M., & Kulkarni, P. Melodic Pattern Recognition and Similarity Modelling: A Systematic Survey in Music Computing. Journal of Trends in Computer Science and Smart Technology, 2022; 4(4): 272-290.

[17] Kreimer, S. This Neuromuscular Specialist Keeps Life Humming with Guitar Playing, Songwriting, and Board Game Development. Neurology Today, 2021;21(21): 17-18.

[18] Kennedy-Macfoy, M. Everything Must Change. European Journal of Women's Studies, 2023; 30(1): 3-6.

[19] Setyaningsih, E., Chandra, I., & William, W. Aplikasi Music Streaming Menggunakan Flutter dilengkapi Music Recognizer. Jurnal Inovasi Teknologi dan Edukasi Teknik, 2021; 1(9): 707-714.

[20] Lee, K. Y., & Hu, C. M. Research on the development of music information retrieval and fuzzy search. Scientific and Social Research, 2022; 4(4): 1-10.

Edited by: Zhigao Zheng
Special issue on: Graph Powered Big Aerospace Data Processing
Received: Oct 18, 2023
Accepted: Nov 6, 2023