

Hybrid Query by Humming and Metadata Search System (HQMS) Analysis over Diverse Features

Nauman Ali Khan

Department of Computer Science, Quaid-I-Azam
University Islamabad, Pakistan.

Mubashar Mushtaq

Department of Computer Science, Quaid-I-Azam
University Islamabad, Pakistan.

Abstract— Retrieval of music content over web is one of the toughest job tasks and found some of the significant challenge. Song retrieval over web is the emerging problem from the category of Music Information Retrieval. Several searching techniques related to metadata and content of song are developed and implemented. In this paper we are going to propose and evaluate a new hybrid technique for song retrieval named as “hybrid query by humming and metadata search system” (HQMS).HQMS is a hybrid model that is based on metadata and query by humming. HQMS narrow downs the relevant results thus improve overall accuracy of the system. Two filters, metadata and query by humming work together and enhances overall accuracy. The system is evaluated song wise, metadata wise and Age wise. Issues related to query reformulation are also highlighted as a part of the paper. More over the evaluation validates the hypothesis and the proposed Architecture.

Keywords- Query by humming; Metadata; MIR; Hybrid system; Pipe and Filter; Evaluation.

I. INTRODUCTION

Validation of any software system is the most crucial phase of software testing to identify the software quality, as it evaluates the validity of the system on the basis of different attributes [1]. In this paper, we have discussed the experimental setup, dataset, development and evaluation of Hybrid Query by Humming and Metadata Search System (HQMS) developed by us [2]. Query by humming is the well known Content Based Searching Technique used for song retrieval. Query by humming takes humming of any song as a query and by applying different pattern matching algorithms, evaluation of relevant retrieve results is performed [3]. HQMS is an improved version of Query by humming techniques; in which we have use metadata as a collaborative part to ordinary QbH technique. Improving the overall accuracy through result filtration is one of the core inspiration of HQMS.

One of the common mechanisms used for the song retrieval is the keyword query based searching mechanism and that totally relies on the metadata of songs. Content inside file is generally considered when either the metadata information stored with is less explained or user is not familiar to the metadata file. Content Based Searching basically searches in all parts of the file and tries to find the best match. When we are considering songs, contents can be electronic waves stored in some specific format. In the field of songs retrieval, Content Based Searching is commonly applied when a user has very less knowledge about a song that is to be searched [4].

If we observe closely, when a user is interested in searching any particular song, it mostly tries to remind the lyrics of the song, if that is not possible, it tries to remind the name of the movie or album, if that is also not possible, he tries to hum the tune of that song. QbH was introduced to provide the facility to search by Humming. Our system (HQMS) of handling some critical issues which ordinary QbH (Query By Humming) cannot resolve. HQMS is a hybrid system that searches the most relevant list of songs on the basis of metadata and humming. HQMS provides better results in the term of accuracy as compare to ordinary QbH.

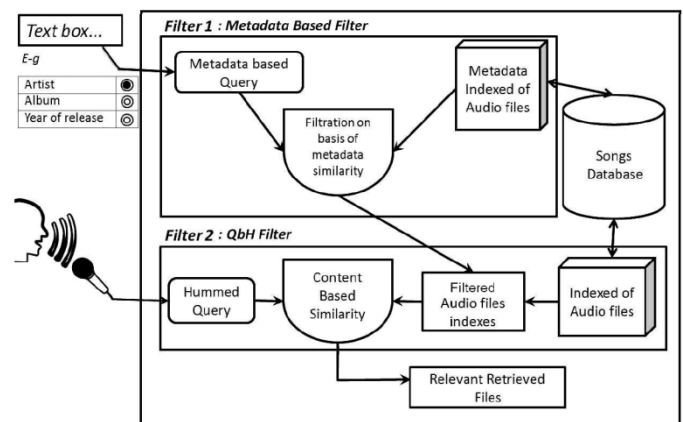


Figure. 1. Architecture of HQMS [2]

In HQMS System ordinary query by humming system was modified by an extra component of metadata. Fig. 1 explains the Architecture of HQMS. Two filters, metadata and query by humming are the core components of HQMS. Reason of introducing two filters in series is to decrease the size of dataset and to increase the overall accuracy of relevant retrieved results. The main objective of First Metadata Filter is to reduce the resultant dataset so that Second Query by Humming Filter can efficiently and accurately find the relevant results. Metadata based filter sort out the list of songs by Artist Name, Album Name, Year, Genre and Title whereas second filter only matches those files which are being filtered by first filter. In the HQMS it was described that two filters can provide better relevant files and the evaluations of HQMS given in this paper.

Basic motivation of the HQMS was that when a user is interested in finding a song when it do not know the exact lyrics of song but remember the tune and some portion of the metadata; in such case HQMS come across with better results

than the ordinary QbH. Both Filters work with collaborated effort.

The rest of the paper is arranged as follows: Section II describes the literature review on query by humming, metadata and hybrid systems. Section III elaborates the HQMS in detail. Section IV discusses the experimental setup for the HQMS. Section V demonstrates development of HQMS. Section VI explains the extensive evaluation of HQMS in different directions. In Last section VII concludes with future directions.

II. LITRATURE REVIEW

Searching activity is always considered as the common activity of the end user. Various searching techniques are invented, developed and implemented. Music searching activity inside computer or over web is the sub activity of searching category. Field of searching music from large repository of internet is called as Music Information Retrieval (MIR). MIR is a vast sub field of Information Retrieval (IR) especially dedicated to resolve the issues during retrieval of music files. The process of MIR generically depends on either metadata or the content of song [5].

Metadata of song works as a main core for MIR. Several metadata based searching techniques are made like vector space model, Boolean model, indexing, invert index file, cosine measure etc [6] [7] [8] [9]. Currently storing of exact metadata is being more emphasized as per to the standards of Web 2.0. According to the standard that sort of metadata is preferred which have some meaning that somehow defines the proper context of song. As metadata of song is less as compared to ordinary document files so different string mining techniques are also applied [5].

For quick retrieval existing IR indexing and clustering techniques are also applied just to handle such critical issues associated to the Music Retrieval [9] [10]. Searches get enormous when we are dealing with huge databases of songs

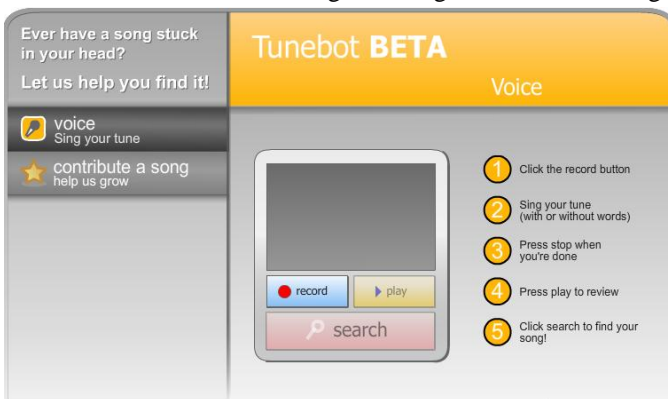


Figure 2. Tunebot System [11]

[5]. Some of the existing Query by humming systems are explained below.

TUNE BOT is National Science Foundation funded project made by a Northwestern University Interactive Audio Lab [11]. TUNE BOT is basically a search engine that works on the basis of query by humming. System contain total of 4990 songs. Each song in the repository is linked to Amazon online

shopping store. Genetic Algorithm is used to improve the overall accuracy. System improves performance by adopting explicit feedback from user. Whole system works in three steps, firstly takes the query from user as humming, then shows a list of result to user and ask whether it satisfies the need or not. By taking feedback training of Genetic Algorithm is performed and performance is improved accordingly [11] [13].

Midomi is the commercial website developed by Me-lodis

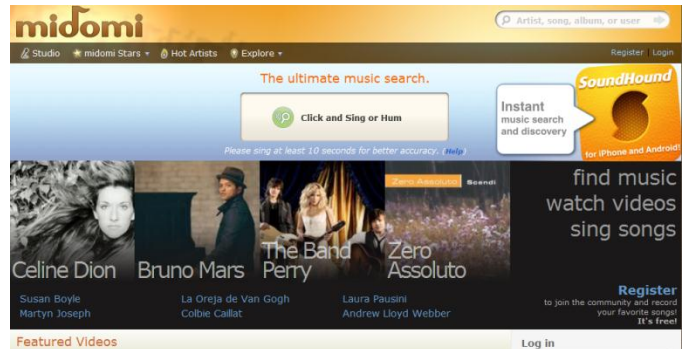


Figure 3. Midomi System [14]

Corporation in August 15, 2006 [14]. Midomi works on query by humming, singing and whistle also provide an advance search on the basis of genre and language. Midomi got the huge repository of more than two million music files.



Figure 4. Soundhound System [15]

Sound hound is another project of Melodis Corporation specially designed for mobile and iPods [15]. Sound Hound is also commercial software works same as mi-domi on basis of hum, sing and whistle. Currently Sound Hound is designed for iphones, ipod touch, iPods and android mobiles [15].

Musipedia is an open source project specially focusing on retrieving songs in very diverse manner [16]. For the ease of end user, system provides a facility to search song in three different categories of Melodic transcription, Contour transcription, Rhythm based. System supports query by humming, whistling, singing, contours and tapping.

Music and Audio Retrieval Tools (MaART) is open source project [17]. MaART is basically a combination of different

software components specially made to evaluate different systems. In majority, system is written in C++ language. Few of the main features of MaART is extracting text from midi file, converting of any midi audio file into piano notes and extracting part from any wav file.

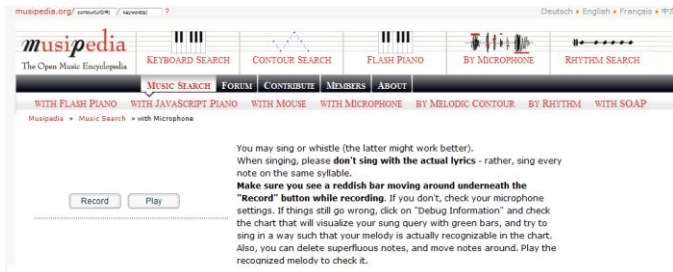


Figure. 2. Musipedia System [16]

Sloud is an ActiveX technology based project [18]. Indexing of audio files and query is the core theme of this project. Sloud uses client and server architecture. Client side user query is transformed into segments and then matched to the existing indexes of stored repository. Pitch recognition measure is used for matching.

Music contents over web are multiplying day by day [5]. Currently the greatest challenge is to reduce the relevant retrieve results for the end user. Metadata based searching technique gets flop when a user is unable to provide any metadata query. To overcome this limitation Content based searching techniques plays a vital role. Content based technique look into the audio file by extracting features. Content and metadata based searching techniques works side by side. A content based searching technique enhances the overall performance of metadata search system rather than a full replacement.

III. PROPOSED ARCHITECTURE

Results of information Retrieval System can be more accurate if detailed query is provided to a search system. This section describes our proposed HQMS. In our system, filtration feature acts as a core to the pipe and filter software architecture style thus resulting in a "Hybrid QbH and Metadata Search System" shown in Fig.1. Pipe and filter software architecture style is mostly used when output of one filter is considered as the input of second filter. In the commonly adopted approaches, there is a single filter having single input and output unit [12].

In proposed architecture, we present two filters in series, such that search query has to go from the pipe that has to pass from both filters one by one. The first filter is metadata wise search and the second follows a Query by Humming process as shown in Fig [2].

Architecture strength lies in the base concepts for building as per needs of user. Behind the adoption of a series pipe and

filter architecture, is the vital goal to de-crease the number of files to be checked by a QbH system.

A. Metadata Filter

Metadata filter is actually the combination of multiple sub-filters which work altogether to get precise results. User has the privilege to provide any sort of information that has the probability of presence in metadata. Metadata information can be related to any of the attribute of the song's metadata. The provided information would be matched to provide a set of possible outcomes. The filter will be operated against the inputted query in the metadata query section. But if a user does not input any metadata than Hybrid system will work as ordinary QbH system, as this filter would not provide an output. This system is capable of handling multiple metadata queries simultaneously.

According to the survey it was examined that most of the users search song on the basis of artist name and title of the song [24], by this we conclude that there is equal possibility that a user wants to give more than one inputs as a query. To handle such situation first filter i.e., Metadata based filter will work as a multiple sub-filters. It will further reduce the number of files for the next filter.

B. Query by humming Filter

As soon as the results are being produced by the first filter (metadata), they are stored in a temporary repository. This repository acts as an input unit for the second filter (QbH) along with the hummed query from the end user. Hummed query is then transformed into time series graph and then that graph is matched accordingly to the pre-indexed files. The filtering process of this unit is to provide further filtration to the temporary repository on the basis of hummed query matched. The match that is found in temporary repository is to be further purified one by one by humming comparison.

Thus, looking at the architecture on a broader perspective, first filter concise the scope of search and second filter increases the relevance of outcome with the desired query from user.

The outcome shown after the processing of both the filters indicates the relevant retrieved files against a query. Other than simple retrieval of relevant audio files, the system can also work as a strong copyright violation detector.

The system would work for both metadata and hum, thus will provide a more authentic result. There have been many issues where a copyrighted song is sung again by a person which violates the copyrights of the owner. Our system can be used in detecting such violations and privacy; copyrights checking authorities and system like [22], will have a prominent advantage of using it.

IV. EXPERIMENTAL SETUP

We Gathered a Dataset of total 150 Pakistani songs in six different languages, such that 25 songs for each language and 25 English of songs. Seven selected languages were Urdu, Sindhi, Punjabi, Pashto, Balochi, Kashmiri and English. After applying Stratified Random Selection five songs from each category were selected as a dataset for the Evaluation.

TABLE I
DETAILS OF DATASET OF SONGS

Language	No. of Songs
Urdu	5 Songs
Punjabi	5 Songs
Pashtu	5 Songs
Sindhi	5 Songs
Kashmiri	5 Songs
English	5 Songs
Balochi	5 Songs

Description of Dataset is described in the Table I.

Three different age users were selected for six categories of Pakistani (Cultural / Regional) languages i-e Urdu, Pashto, Punjabi, Balochi, Kashmiri, and Sindhi. Among three different categories, age of first category was in between 10year to 15year, age of Second category was in between 16year to 20year and age of Third category was in between 21year to

TABLE II
DETAILS OF USERS

Language	Age limit >=11yr and <=15yr		Age limit >=16yr and <=20yr		Age limit >=21yr and <=25yr	
	1 Male	1Female	1 Male	1Female	1 Male	1Female
Urdu	1 Male	1Female	1 Male	1Female	1 Male	1Female
Punjabi	1 Male	1Female	1 Male	1Female	1 Male	1Female
Pashtu	1 Male	1Female	1 Male	1Female	1 Male	1Female
Sindhi	1 Male	1Female	1 Male	1Female	1 Male	1Female
Kashmiri	1 Male	1Female	1 Male	1Female	1 Male	1Female
Balochi	1 Male	1Female	1 Male	1Female	1 Male	1Female

25year. Details of users are de-scribed in the Table II.

V. DEVELOPMENT OF HQMS

We developed HQMS architecture [2] as shown in Fig.1. Five categories were taken in the consideration while developing metadata filter that includes Album Name, Artist Name, Genre, Title of the Song and Year of Release. We used a Levenshtein string matching function for the string comparison of query and stored metadata of song [19]. By observing the human natural behavior, for searching any song we introduced query weighting as well. Motivation scenario for the giving weight to a query was

“When a user is interested in searching a song, such that he does not remember the lyrics of the song but remembers the tune of the song and little portion of Metadata the song, and in that little metadata he can further classifies the priority to each category”

For example a user is pretty much sure that the song is of Michael Jackson Artist but he is less sure about the genre of the song whether it is Rock or Pop to handle such situation we provided a priority based advance search. User can easily adjust weight of the different parameter of query by using a

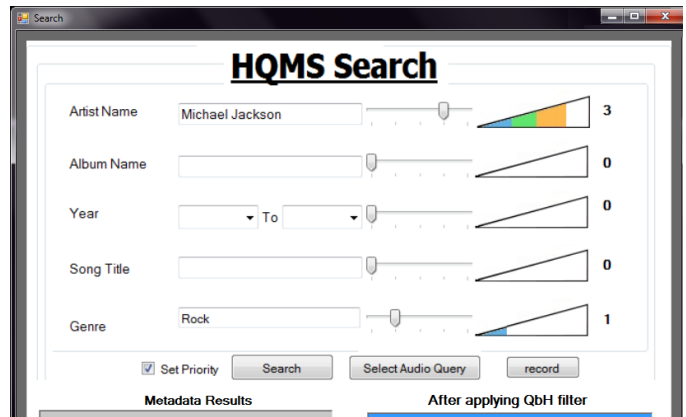


Figure. 6. HQMS System interface for metadata search

slide bar. Between the values of 0 to 5 i-e (0, 1, 2, 3, and 4) were kept. Fig. 6 explains the interface of the priority Based Advance Search.

MySQL was used as repository for storing of metadata of song [20] and, Similarity Matrix values. When a user gives a weighted query, in the backend each category of query compares to the corresponding metadata of each song by using a Levenshtein similarity function [19]. After comparison each values is multiplied by the user adjusted priority value given through the interface. At the end accumulative similarity is calculated. Fig .7 explains the whole mechanism of the metadata filter.

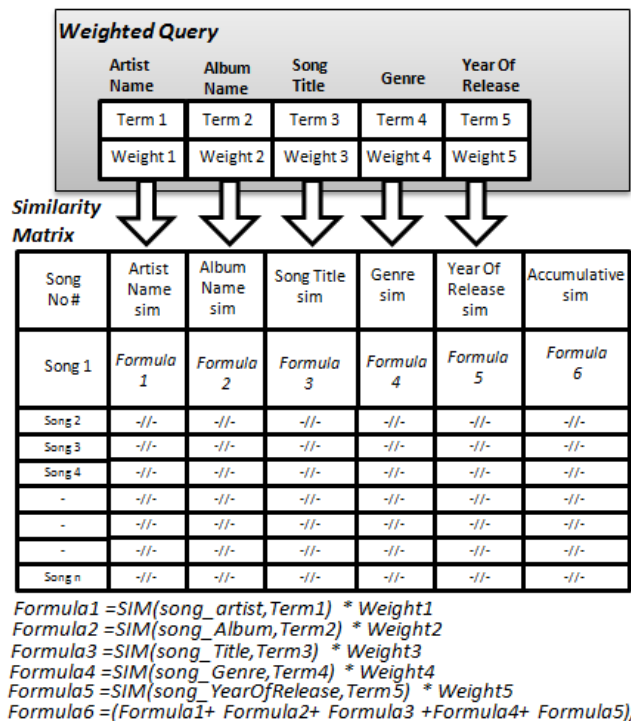


Figure. 7. Mechanism of metadata filter

Usually user finds its relevant result in the top 40% of the total searched results. We shifted the indexes of total of 40% of the total searched results to the second filter of Query By

Humming. 40% of results was actually written in the text file as the input to the second filter.

For the second filter of Query by Humming we used open source project of CompariSong developed in Java [21]. CompariSong firstly converts audio file into segments. 10 segments per second were extracted and on the basis of Fundamental Frequency and Audio Power, numerical time series was converted to letters. Same procedure of extracting and assigning was done on the hummed query as well. For finding the matching between hummed query and pre-indexed file Levenshtein distance measure was used [19]. Relevant retrieved results were presented in a sorted list, such that most relevant at the top.

System used for evaluation was HP ProBook 4520s, Core i5 480M, 4GB Ram, 1GB Graphic card and 500GB Hard Disk. All the user recordings were taken in an empty room almost a noise free room. Earlier discussed dataset was given to each user. They heard each song once and recorded the hum for all the songs.

VI. EVALUATION OF HQMS

One male and one female as a group were selected for each category. Information related to each user was

- Name
- Age
- Gender
- Mother Language
- Region
- Qualification

The main reason to choose a diverse dataset of songs of different languages and diverse user of different ages, regions, languages and genders was just to evaluate whether each category have some effects on the other category or not. We evaluate our system in a very extensive manner and also checked the impact. We made evaluation in the following categories.

1. User Related
 - i. Gender Wise
 - ii. Age Wise
 - iii. Language Wise
2. Metadata Related
 - i. String similarity 0% and records 100%
 - ii. String similarity 20% and records 80%
 - iii. String similarity 40% and records 60%
 - iv. String similarity 60% and records 40%
 - v. String similarity 80% and records 20%
 - vi. Threshold string similarity 60% and records 40%
3. Song Related
 - i. Song Wise Average Accuracy
 - ii. User Wise Average Accuracy

We used Mean and Standard Deviation statistical measure to find the average and difference between the results. For the evaluation of query by humming system we also introduced a new measure of accuracy percentage that is explained in

equation .1.1. N is the number of total songs in repository where P is the position of relevant song in the list of results.

By using this formula we can find the accuracy of Query by humming system on the basis of relevant song position such

$$Accuracy \% = \frac{(N - P) + 1}{N} \times 100 \quad (1)$$

that after searching if the relevant song is at the top of the result list, it means accuracy is 100%.

A. Gender Wise

Females have high pitch as compared to male so initially, we divided whole dataset into two groups of male and female for gender based evaluation and we selected only those solo songs which were sung by either male or female. For each song initially, male group recorded the query then female group recorded the query. Averages of both groups were compared for each song.

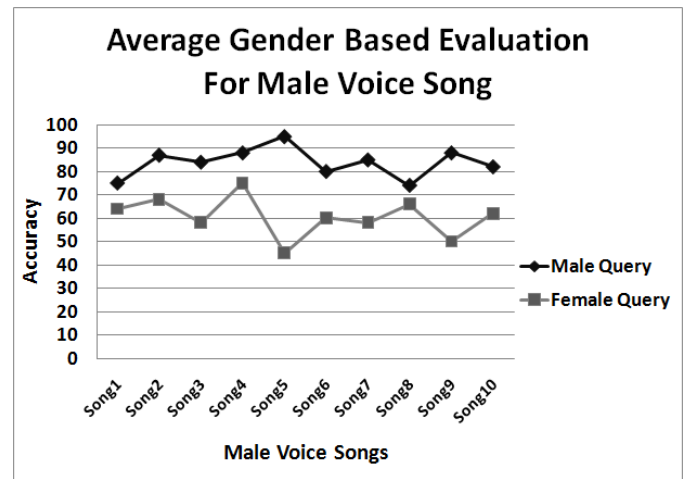


Figure. 8. Average gender based evaluation for male voice song

Fig. 8 shows the head to head comparison of male and female queries for each song. Likewise both set of groups were then given female sung songs and same procedure was adopted for comparison.

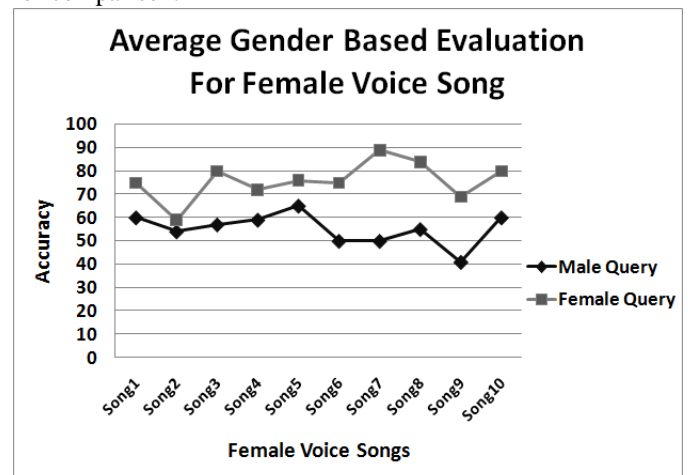


Figure. 9. Average gender based evaluation for female voice song

Fig. 8 shows Average Gender Based Evaluation of the male and female query for Male voice Songs and it was evaluated that male's queries more accurate results as compared to female's queries gave.

Respectively Fig. 9 shows evaluation of both groups for female sung song and it was examined that females can more accurately search female sung song

In graph (Fig. 10) y-axis show the accuracy in percentage and x-axis shows the songs. For each song two values are shown one the average female value and other male average accurate value.

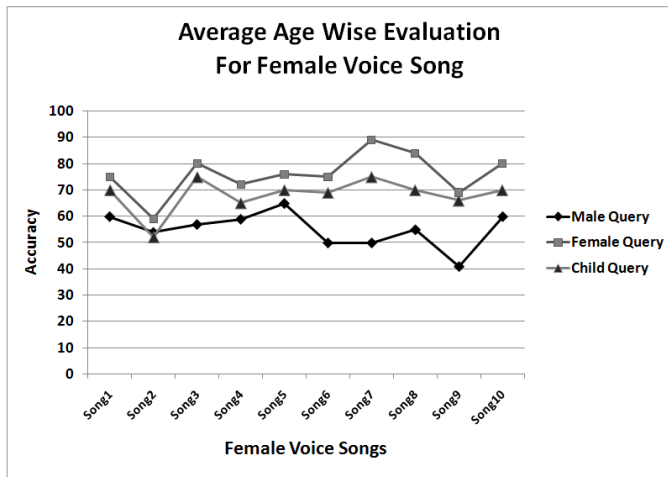


Figure. 10. Age group wise accuracy for female songs

B. Age Wise

As we described earlier in the paper that we divided our total population into three main groups age wise. The ranges of three groups are (11yr to 15yr), (16yr to 20yr) and (21yr to 25yr). The main reason to divide age wise is to evaluate whether accuracy differs with respect to age or not. We got some surprising results for the age of 11yr to 15 yr that user from this age got good accuracy for the female voice songs. The reason which we concluded was that commonly the pitch of small kids is high. High pitch is the common factor to female voice. Fig. 10. specifically shows the comparison of female, male and children group of user.

Fig. 13 show the comparison for each age groups accuracy. Average of each group is considered as a single value for every

song in dataset. Three values for each song depict three average values of each age group.

C. Language Wise

Songs are recorded in different languages. Similarly song listeners also have different mother language and it is the common observation that single language users use to listen multilingual songs. By keeping these facts in mind we evaluated our system diversely. We took songs of six cultural languages of Pakistan and english language songs as dataset. Similarly we selected six different cultural language speaking users. We gave each set of users' full dataset, and directed them to listen and record there. After evaluation we concluded that if single language user sing's its own language song, the accuracy of matching is higher. Fig. 11 shows the overall average values of all language groups.

Language	Songs							
	Urdu	Pashto	Punjabi	Sindhi	Kashimiri	Balochi	English	
Users	Urdu	81.2381	54.7619	68.7619	56.95238	60.28571	44.09524	63.42857
	Pashto	64.38095	79.61905	55.80952	52.09524	51.2381	68.19048	56.38095
	Punjabi	72.95238	49.61905	83.52381	59.42857	61.90476	39.2381	52.09524
	Sindhi	66.38095	54.28571	56.38095	80.85714	59.14286	56.19048	40.28571
	kashimiri	68.09524	61.14286	64.47619	46.47619	75.71429	39.52381	38.19048
	Balochi	63.42857	64.38095	44.66667	44.7619	47.42857	82.57143	45.61905

Figure. 11. Language to language comparison

In Fig. 11. the shaded diagonal values show maximum accuracy. The shaded diagonal values represent the similarity between same language songs and humming. This concludes that if single language users tries to hum or sing a song of its own language then maximum accuracy can be achieved. In this regard certain audio query reformulation or alignment techniques are to be performed so that maximum accuracy can be obtained.

This can be explain by a scenario if language "A" user tries to hum a song recorded in language "B" so in such case accuracy will be less. But if somehow language "A" audio query get transformed into language "B" accent then more relevant results can be obtained. During evaluation of HQMS we also analyzed if any two languages having similarity in regard of accent then query reformulation is not necessary. In our case two cultural languages of Pakistan Pashto and Balochi have similarity in accent as compare to other cultural languages of Pakistan. But audio query reformulation can play an

String similarity 0% and records 100%								
		Songs						
Users		Urdu	Pashto	Punjabi	Sindhi	Kashmiri	Balochi	English
Urdu		81.24	54.76	68.76	56.95	60.29	44.1	63.43
Pashto		64.38	79.62	55.81	52.1	51.24	68.19	56.38
Punjabi		72.95	49.62	83.52	59.43	61.9	39.24	52.1
Sindhi		66.38	54.29	56.38	80.86	59.14	56.19	40.29
Kashmiri		68.1	61.14	64.48	46.48	75.71	39.52	38.19
Balochi		63.43	64.38	44.67	44.76	47.43	82.57	44.57

String similarity 20% and records 80%								
		Songs						
Users		Urdu	Pashto	Punjabi	Sindhi	Kashmiri	Balochi	English
Urdu		81.66	56.82	72.15	59.87	63.39	47.39	66.63
Pashto		67.13	81.76	59.58	56.06	53.2	70.82	60.34
Punjabi		76.75	52.66	84.9	62.92	64.82	41.96	55.39
Sindhi		70.08	57.7	60.46	82.87	63.49	58.44	47.68
Kashmiri		70.46	63.51	66.94	48.56	77.71	42.92	41.3
Balochi		66.75	68.85	46.85	48.08	49.7	83.13	49.68

String similarity 40% and records 60%								
		Songs						
Users		Urdu	Pashto	Punjabi	Sindhi	Kashmiri	Balochi	English
Urdu		83.05	58.1	73.43	61.14	64.67	48.67	67.91
Pashto		68.38	83.14	60.86	57.33	54.48	72.1	61.62
Punjabi		78	53.91	86.1	64.19	66.1	43.24	56.67
Sindhi		71.33	58.95	61.71	84.48	64.76	59.71	48.95
Kashmiri		71.71	64.76	68.19	49.81	79.14	44.19	42.57
Balochi		68	70.1	48.1	49.33	50.95	84.86	50.95

String similarity 60% and records 40%								
		Songs						
Users		Urdu	Pashto	Punjabi	Sindhi	Kashmiri	Balochi	English
Urdu		89.05	61.21	76.54	64.25	67.78	51.78	71.02
Pashto		71.66	88.19	63.97	60.45	57.59	75.21	64.73
Punjabi		81.27	57.18	90.1	67.3	69.21	46.35	59.78
Sindhi		74.61	62.23	64.99	87.95	67.87	62.83	52.06
Kashmiri		74.99	68.04	71.47	53.08	86.77	47.3	45.68
Balochi		71.27	73.37	51.37	52.61	54.23	89.98	54.06

String similarity 80% and records 20%								
		Songs						
Users		Urdu	Pashto	Punjabi	Sindhi	Kashmiri	Balochi	English
Urdu		95.15	64.32	79.65	67.37	70.89	54.89	74.13
Pashto		74.47	95.3	67.08	63.56	60.7	78.32	67.84
Punjabi		84.09	60	96.2	70.41	72.32	49.46	62.89
Sindhi		77.43	65.04	67.81	94.05	70.98	65.94	55.18
Kashmiri		77.81	70.85	74.28	55.9	91.88	50.41	48.79
Balochi		74.09	76.19	54.19	55.43	57.04	95.08	57.18

Threshold String similarity 60% and records 40%								
		Songs						
Users		Urdu	Pashto	Punjabi	Sindhi	Kashmiri	Balochi	English
Urdu		89.05	61.21	76.54	64.25	67.78	51.78	71.02
Pashto		71.66	88.19	63.97	60.45	57.59	75.21	64.73
Punjabi		81.27	57.18	90.1	67.3	69.21	46.35	59.78
Sindhi		74.61	62.23	64.99	87.95	67.87	62.83	52.06
Kashmiri		74.99	68.04	71.47	53.08	86.77	47.3	45.68
Balochi		71.27	73.37	51.37	52.61	54.23	89.98	54.06

Figure. 12. Metadata based evaluation

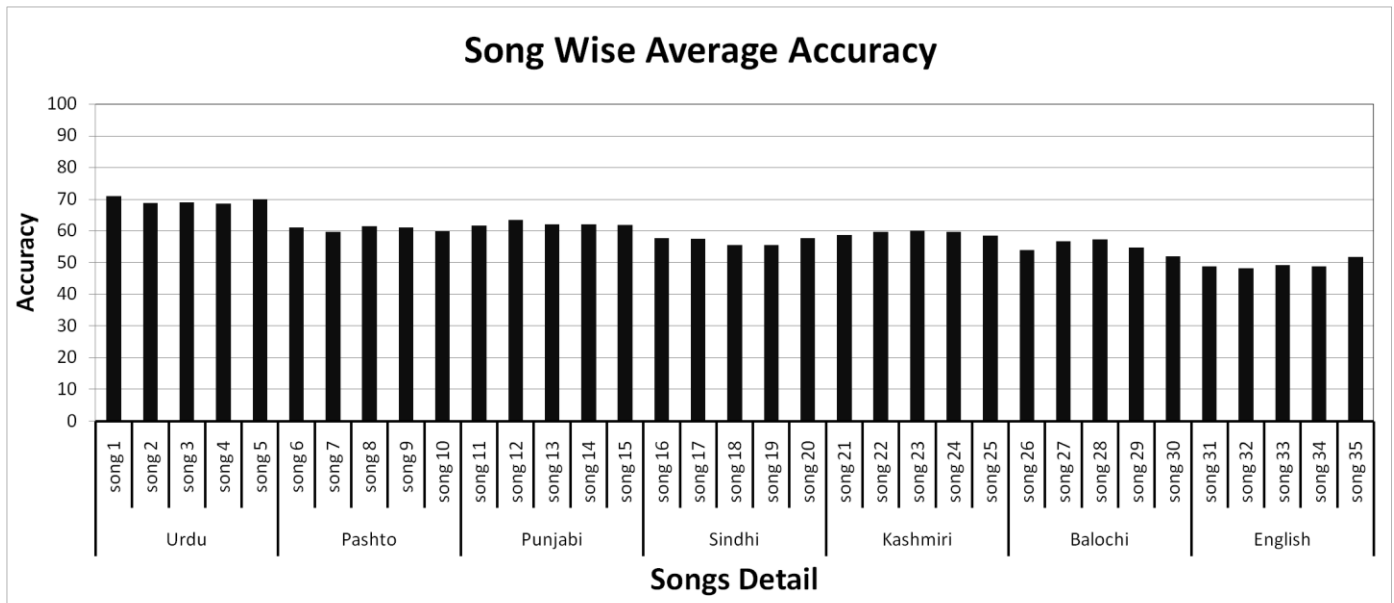


Figure. 13. Song wise average accuracy

effective role when two languages have different accents.

D. Metadata Evaluation Results

For the evaluation of metadata we tested values on different intervals and suggested threshold. We made 5 intervals for evaluation as we have limited dataset. Removal of biasness from results was the main objective for creating intervals. We choose 5 intervals on the basis of query similarity and filtration of records. Initially we gave dataset to all users to listen and then hum. If in this case we could have also asked them to give metadata of songs as well then there was a chance of biasness because by listening song one can understand lyrics and get closer to exact title. In HQMS case, two parameters are associated to metadata filter i.e. (similarity value between query and relevant song metadata) and (percentage of filtration required accordingly).

We distributed above mentioned parameter into 5 groups

- String similarity 0% and records 100%
- String similarity 20% and records 80%
- String similarity 40% and records 60%
- String similarity 60% and records 40%
- String similarity 80% and records 20%
- Threshold string similarity 60% and records 40%

Fig. 12. shows results for each category. We also suggested threshold i.e. String Similarity 60% and record 40%. Whenever a user is interested in searching a song then the query which he gives for searching is almost 60% similar to the relevant song. Secondly 40% of record shows the percentage of filtration that means we selected top 40% of total record after applying string similarity.

Fig. 12. shows 5 intervals values and 1 threshold value. As discussed earlier shaded diagonal values shows higher accuracy

because of same language. The overall accuracy improves when string similarity value gets increased and filtration gets decreased. The ratio of similarity and filtration selection can be dynamic by taking explicit values from user.

E. Song Wise Average Accuracy

Songs are recorded with a variety of compositions and in different genres too. Some songs are easy to sing or hum but some songs are not memorized very quickly. To observe such variation we evaluate accuracy for each song and found that in user population group mostly user can hum more accurately for Urdu language songs. Two reasons which we concluded were, Pakistan National language is Urdu and in our population group majority of users were literate which mean they were quite familiar with Urdu language, and most of the users were having Urdu language as the medium of communication.

Fig. 13 shows the overall comparison of each song with respect to all users. Individual accuracy of song is measure just to check the song is easy to sing or hum and vice versa.

F. User Wise Average Accuracy

It is a common observation that songs are recorded in different languages and commonly one language user use to listen different language songs. To check this idea we took a small survey.

We distributed whole population into groups of language then each group was further classified into age group and gender as shown in Fig. 14. In this ``F" represent Female and ``M" represent Male.

According to the evaluation, age group between 20yr to 25yr has shown more accurate results as compared to rest of the group. The reason which we concluded was that users belonging to this age group are mature, and they can hum or sing more accurately as compared to other age groups.

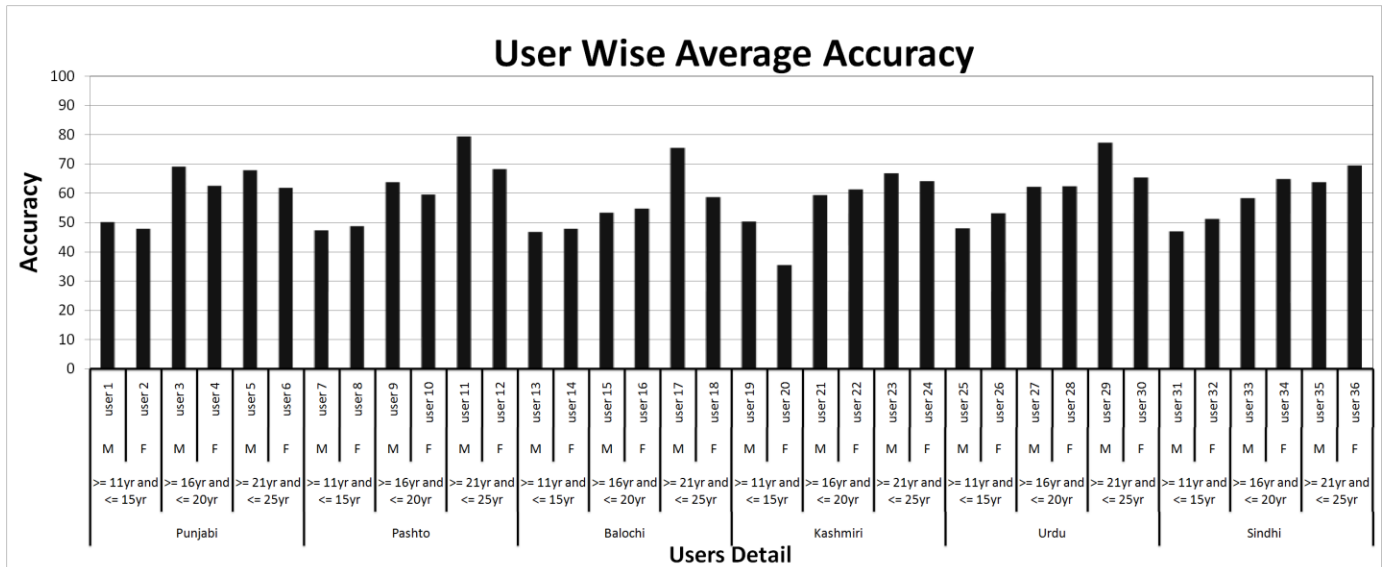


Figure. 14. User wise average accuracy

VII. CONCLUSION

Hybrid Query by Humming and Metadata Search System (HQMS) is the proposed model of search engine for songs [2]. This hybrid model works on the two parameters of songs i.e. metadata and content. Pipe and Filter software architecture style is used to express the model. In this paper, we implemented the proposed HQMS model and evaluated results by considering different parameters. HQMS includes two main filters of Metadata and Query by Humming. For evaluation we took sample of 36 users and 35 songs from different Cultural Languages of Pakistan. Stratified random sampling method was adopted. Evaluation of HQMS was done in three generic categories of User Related, Metadata Related and Songs Related. In future, we aim to enhance HQMS by considering different parameters and matching algorithms related to both metadata and contents of song.

ACKNOWLEDGMENT

The authors would like to appreciate the work and time given by G. Dzhabazov, developer of CompariSong, we are specially thankful to Danish Saif Talpur and Mudasser Ali Khan for their valuable time they served for reviewing the paper.

REFERENCES

- [1] G.J. Myers. The art of software testing. Wiley-India, 2008.
- [2] N.A. Khan, M.A. Khan, and M. Mushtaq. Hybrid query by humming and metadata search system (HQMS). Proceedings of the 8th International Conference on Frontiers of Information Technology, page 9, 2010.
- [3] A. Ghias, J. Logan, D. Chamberlin, and B.C. Smith. Query by humming: musical information retrieval in an audio database. In Proceedings of the third ACM international conference on Multimedia, pages 231–236. ACM, 1995.
- [4] Y.H. Tseng. Content-based retrieval for music collections. In Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, pages 176–182. ACM, 1999.
- [5] M.A. Casey, R. Veltkamp, M. Goto, M. Leman, C. Rhodes, and M. Slaney. Content-based music information retrieval: current directions and future challenges. Proceedings of the IEEE, 96(4):668–696, 2008.

- [6] N.C. Maddage, H. Li, and M.S. Kankanhalli. Music structure based vector space retrieval. In Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval, pages 67–74. ACM, 2006.
- [7] N. Kosugi, Y. Nishihara, S. Kon`ya, M. Yamamuro, and K. Kushima. Music retrieval by humming-using similarity retrieval over high dimensional feature vector space. In Communications, Computers and Signal Processing, 1999 IEEE Pacific Rim Conference on, pages 404–407. IEEE, 1999.
- [8] L. Chen and B.G. Hu. An implementation of web based query by humming system. In Multimedia and Expo, 2007 IEEE International Conference on, pages 1467–1470. IEEE, 2007.
- [9] C. Francu and C.G. Nevill-Manning. Distance metrics and indexing strategies for a digital library of popular music. In Multimedia and Expo, 2000. ICME 2000. 2000 IEEE International Conference on, volume 2, pages 889–892. IEEE, 2000.
- [10] L. Lu, H. You, and H.J. Zhang. A new approach to query by humming in music retrieval. In Proceedings of the IEEE International Conference on Multimedia and Expo. Citeseer, 2001.
- [11] D. Little, D. Raffensperger, and B. Pardo. A query by humming system that learns from experience. 2007.
- [12] D. Garlan and M. Shaw. An introduction to software architecture. (CMU-CS-94-166), January 1994
- [13] D. Little, D. Raffensperger, and B. Pardo. Online training of a music search engine. 2007.
- [14] midomi. midomi. URL: <http://www.midomi.com/>.
- [15] soundhound. soundhound. URL: <http://www.soundhound.com/>.
- [16] musipedia. musipedia. URL: <http://www.musipedia.org/>.
- [17] MaART. Maart. URL: <http://maart.sourceforge.net/>.
- [18] V. Lobaryev, G. Sokolov, and A. Gordeyev. Sloud query-by-humming search music engine. 2006.
- [19] E.S. Ristad and P.N. Yianilos. Learning string-edit distance. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 20(5):522–532, 1998.
- [20] AB MySQL. MySQL: the world’s most popular open source database. MySQL AB, 2005.
- [21] G. Dzhabazov. Comparisong: Audio comparison engine. International Book Series Number 11, page 33, 2009.
- [22] K.-H. Kim, M. Lim, and J.-H. Kim. Music copyright protection system using fuzzy similarity measure for music phoneme segmentation. pages 159{164, 2009.
- [23] Y. Song, Y. Wang, Y. Ju, M. Seltzer, I. Tashev, and A. Acero. Voice search of structured media data. pages 3941{3944, 2009.

AUTHORS PROFILE

Nauman Ali Khan is M.Phil candidate in the Faculty of Computer Science, Quaid-I-Azam University Islamabad, Pakistan. He received his BS(CS) honors Degree from N-W.F.P University of Engineering & Technology Peshawar. His research interests include Music Content retrieval, Query by Humming, and multimedia Searching.

Mubashar Mushtaq is an Assistant Professor of P2P multimedia Video Streaming at the Faculty of Computer Science, Quaid-I-Azam University Islamabad, Pakistan. Mubashar Mushtaq did PhD from CNRS LaBRI Lab—University of Bordeaux 1. He has received his MSc and MS degrees in Computer Science from Quaid-I-Azam University, Islamabad, Pakistan and University of Bordeaux 1, France in 2002 and 2004, respectively. His research interests include Multimedia Streaming over next generation networks, QoS, mobile IP-TV, and multimedia caching.