Jam with Jamendo: Querying a Large Music Collection by Chords from a Learner's Perspective

Anna Xambó
Centre for Digital Music
Queen Mary University of London
a.xambo@qmul.ac.uk

Johan Pauwels
Centre for Digital Music
Queen Mary University of London
j.pauwels@qmul.ac.uk

Gerard Roma
CeReNeM
University of Huddersfield
g.roma@hud.ac.uk

Mathieu Barthet
Centre for Digital Music
Queen Mary University of London
m.barthet@qmul.ac.uk

György Fazekas
Centre for Digital Music
Queen Mary University of London
g.fazekas@qmul.ac.uk

ABSTRACT

Nowadays, a number of online music databases are available under Creative Commons licenses (e.g. Jamendo, ccMixter). Typically, it is possible to navigate and play their content through search interfaces based on metadata and file-wide tags. However, because this music is largely unknown, additional methods of discovery need to be explored. In this paper, we focus on a use case for music learners. We present a web app prototype that allows novice and expert musicians to discover songs in Jamendo's music collection by specifying a set of chords. Its purpose is to provide a more pleasurable practice experience by suggesting novel songs to play along with, instead of practising isolated chords or with the same song over and over again. To handle less chord-oriented songs and transcription errors that inevitably arise from the automatic chord estimation used to populate the database, query results are ranked according to a computational confidence measure. In order to assess the validity of the confidence ranked system, we conducted a small pilot user study to assess its usefulness. Drawing on those preliminary findings, we identify some design recommendations for future applications of music learning and music search engines focusing on the user experience when interacting with sound.

CCS CONCEPTS

• **Applied computing** → **Sound and music computing**; • **Social and professional topics** → *Computing education*; • **Human-centered computing** → User studies;

KEYWORDS

user study, query-by-chord, Creative Commons, music information retrieval

AM'18, September 12–14, 2018, Wrexham, United Kingdom

ACM ISBN 978-1-4503-6609-0/18/09…$15.00
https://doi.org/10.1145/3243274.3243291

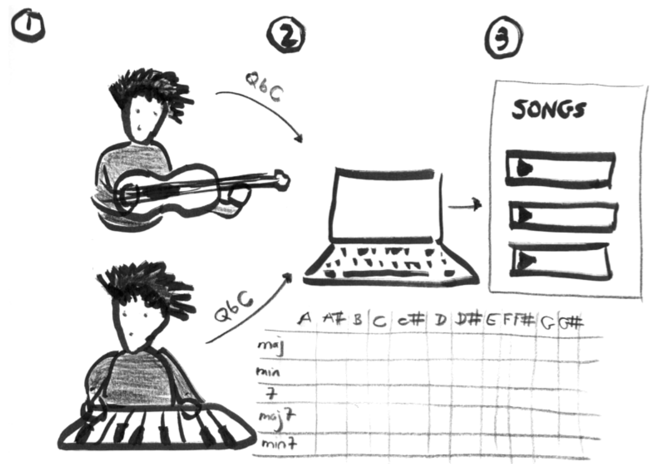


Figure 1: Workflow of our QbC system.

ACM Reference Format:
Anna Xambó, Johan Pauwels, Gerard Roma, Mathieu Barthet, and György Fazekas. 2018. Jam with Jamendo: Querying a Large Music Collection by Chords from a Learner's Perspective. In *Audio Mostly 2018: Sound in Immersion and Emotion (AM'18), September 12–14, 2018, Wrexham, United Kingdom.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3243274.3243291

1 INTRODUCTION

When learning to play a musical instrument by following the traditional path, a fulfilling practice is to play songs that contain a set of chords that the music learner knows. This approach allows the student to apply the learning competences to music genres of her or his interest. It also helps to promote the intention to persist in music education because giving choices to the students (e.g. making musical decisions, developing musical skills in the real world) has been found to affect positively student motivation over the music learning experience [17]. There are a number of music sheet books that are designed for this learning activity, however the set of songs and chords involved are typically limited and, generally, in printed version. The advent and popular use of the Internet and Creative Commons (CC) licenses in the 2000s [9] has promoted the development of a number of online music databases and services (e.g. Jamendo,[1] ccMixter),[2] which offer a large amount of music available. Although it is possible to explore and play the songs from these online databases through music search engines and playlists, to our knowledge there is still a gap of bringing these large collections of music adapted to the music learner needs, which is the topic of interest of this paper.

Here, we present *Jam with Jamendo*, a web-based proof-of-concept prototype focusing on the use case of music. The intention of the system is to facilitate the practice of suitable songs from known chords. The prototype allows novice and expert musicians to explore sounds from the music database Jamendo based on query-by-chords (QbC) from a musical instrument-agnostic approach (see Figure 1). Jamendo is a digital music content provider that provides access to around 200K royalty free music tracks from independent artists, which is suitable for our use case. This research is informed and informs the European funded project AudioCommons [5], which aims at building an ecosystem of CC content, tools and users. Our system allows the music learner to search songs based on a set of chords, which are ordered by confidence measures' criteria. The confidence measures are implemented from an algorithm developed by Pauwels et al. [13] that presents confidence measures applied to a chord estimation task. This research is also inspired by Font and Serra's work on confidence measure applied to tempo estimation in loops [6]. We conducted a small pilot user study in order to assess the validity of the confidence measures' ranked system applied to chord estimation in *Jam with Jamendo*. We were mostly interested in surveying novice musicians who are learning to play a musical instrument. Our research question was to assess the usefulness of a system that suggests a curated list of songs based on a confidence measure metric and retrieved from a particular search of a set of chords. From the preliminary findings of the study, we identify and draw on design recommendations for future applications of music learning and music search engines focusing on the user experience when interacting with sound.

2 BACKGROUND

In music information retrieval (MIR) research, content-based music search engines have been developed as an interesting application for both researchers and users (e.g. [3, 7, 8, 19]. The use of 2D and 3D graphics to display large collections of sound and music databases has been popular [16]. However, to our knowledge, there is little research on the particular task of query by chords. Frequent chord progression patterns in a large audio music dataset have been analyzed by Barthet et al. [2]. There are also online apps that implement chord detection in real time, such as Yanno.[3] TheoryTabs[4] is a web app designed for learning the music theory behind a song showing information about the melody and chord progressions, and allowing to explore transpositions of chords.

There are a number of online music recommendation systems for music learning. Hotttabs [1] is a music aid system for guitar players. The system presents song clustering based on chords retrieved from Echo Nest and YouTube, where it shows the main guitar tabs clustered by difficulty accompanied with video tutorials, however it is unrelated to chord content analysis. Chordify [4] is a web-based app that automatically transcribes chord labels of a provided song, in real time, and allows to play it with chord-based visual feedback. Yousician[5] is a music learning platform, especially designed for guitar players, that gives real time feedback on accuracy and time, where it is possible to upload and follow selected songs. Songle[6] is another web app that provides a real time visualization of the structure and content of the song including beat, melody and chords. The interface of this system is complex and it is not possible to retrieve by your own chord sequences. The novelty of our approach is that it centers on the retrieval task instead of the playback task and generalizes to an instrument-agnostic approach.

[1]http://jamendo.com
[2]http://ccmixter.org
[3]http://yanno.eecs.qmul.ac.uk
[4]https://www.hooktheory.com/trends

3 THE SYSTEM

3.1 Concept

Jam with Jamendo is a web app prototype of QbC that helps the music learner to play songs with a musical instrument. The system suggests a curated lists of songs from the online music database Jamendo. In this way, music learners can adapt the music content to their knowledge and their own learning pace. The web app is aimed at music practitioners, from beginners to experts, who already play an instrument (e.g. guitar, piano, bass). The benefits of using an online app is that it can be accessed from different locations and from multiple types of devices (e.g. laptop, tablet, smartphone). The possibility of retrieving songs from Jamendo based on QbC provides a use case of accessing digitalized CC content that is generally unavailable in traditional music shops. The discovery of new bands is possible. At the same time, there is a potential need of new graphical score representations that can adapt to these new platforms and content (e.g. interactive scores, graphical chord sequence representations). From the two potential scenarios of retrieval and chord progression validation, as a first step we focus here on the retrieval scenario.

3.2 Algorithm

A notable characteristic of the Jamendo database is that it is not curated from a chords point of view, such that some of its content does not contain chords at all. We argue that (annotated) datasets regularly used for the evaluation of audio chord estimation, such as the Isophonics set [10], suffer from a strong case of curation bias. After all, nobody will invest significant effort in annotating a music piece with chords unless they know the piece to contain chords beforehand. This is even more the case for Jamendo than for other large collections, such as Spotify, Deezer and Apple Music, because any individual can upload their tracks for free. There is no music label to act as a filter or financial barrier to entry. Consequently, Jamendo contains a multitude of pieces for which a chord annotation does not really make sense, because the material is monophonic, non-harmonic, based on a non-Western tuning system or simply not of a musical nature.

In order to gracefully handle non-chordal content, we not only need a chord estimation algorithm to enrich the audio database with chord transcriptions, but also a confidence measure that indicates the quality of each transcription. A low confidence should

[5]https://yousician.com
[6]http://songle.jp

be assigned both to pieces for which it is likely that the transcription algorithm has made a mistake and pieces for which a chord annotation is not sensible at all.

For this reason, we use the chord estimation system recently proposed by Pauwels et al. [13], of which an implementation is freely available online.[7] We use their Pointwise Path Difference (PPD) confidence measure, which compares the results of two different ways of decoding a hidden Markov model and derives a confidence value ranging from 0 to 1 from it. Strong agreement between a maximum a posteriori decoder and a pointwise maximum a posteriori decoder leads to a high confidence. This confidence value indicates the quality of the automatic chord estimation for the entire file. It has been shown to correspond well to the numerical evaluation typically used for assessing chord estimation, like in MIREX[8], but no perceptual evaluation of the confidence measure has been performed to our knowledge.

3.3 Dataset

To be specific, our system is built around the catalog of Jamendo Licensing,[9] the entity that aims to license content on Jamendo for commercial purposes, such as in-store radios and background music for films and games. Users can opt in to this service when uploading their content to Jamendo. This catalog contains about 200K tracks, and their audio and metadata are available through an API.[10] Jamendo Licensing deems about 100K of these tracks to be of sufficient quality to actively promote them to its commercial customers. This curation is purely decided on production quality, as opposed to content quality, and removes bad recordings, copyrighted and joke submissions.

Of these 100K tracks, 99 960 had their audio available through the API at the time of constructing our local audio set, which is the set we refer to as the Jamendo dataset for simplicity reasons. We ran the aforementioned algorithm on each of these offline files with a chord vocabulary of 60 chords (maj, min, 7, maj7 and min7 chord types for all 12 pitch classes). The resulting estimated chord sequences and corresponding confidence values were stored in a new database.

In order to get a better idea of the content in the database, we extracted some basic statistics. First we took a look at the distribution of the track durations, displayed in Figure 2. As can be expected of a dataset of this size, the distribution is nearly Gaussian, with a mean of 240 ± 150 seconds and the lower tail naturally clipped at zero. The distribution of confidences, shown in Figure 3, is similarly Gaussian, with a mean of 0.70 ± 0.12.

Finally, we explored the occurrence of chords in the database. Figure 4a shows the percentage of files that contain a chord for our vocabulary of 60 chords. The average number of distinct chords per file is 13.11 ± 7.44. Unsurprisingly, the most popular chords are Cmaj and Gmaj, each contained in 52% of the files. Remarkable is the high number of files that contain min7 chords, especially Amin7, Dmin7 and Emin7, which all appear more often than their corresponding minor triads. Also, the least common chord, Ebmin is surprisingly still present in 5.9% of the files.

[7]https://github.com/jpauwels/chord-estimation-confidence
[8]http://www.music-ir.org/mirex/wiki/MIREX_HOME
[9]https://licensing.jamendo.com
[10]https://developer.jamendo.com

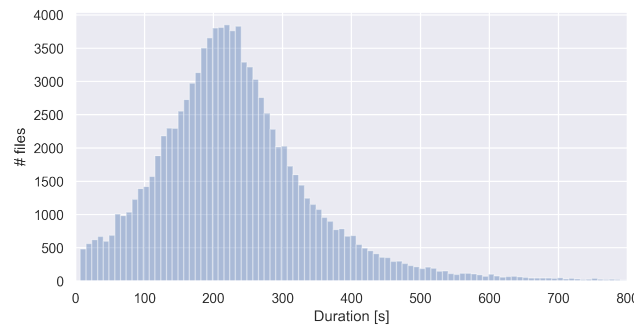


Figure 2: Distribution of the duration of the music pieces in the Jamendo dataset.

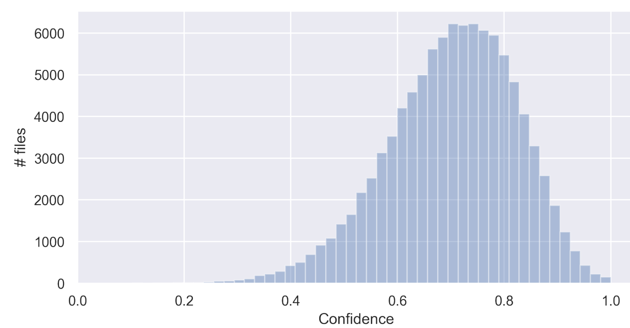


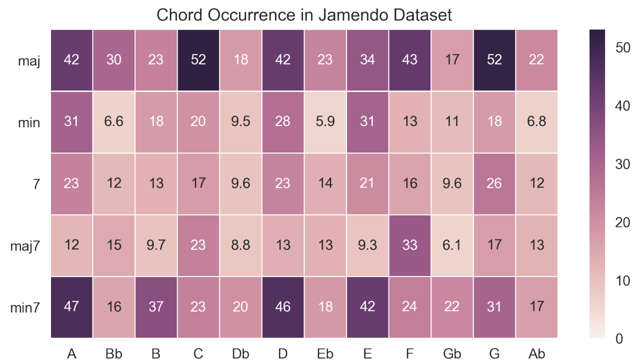
Figure 3: Distribution of the confidence value indicating the quality of the automatic chord estimation of the music pieces in the Jamendo dataset.

We suspect that these statistics are partly skewed because of the inevitable errors of the automatic chord estimation. This is amplified by the fact that in order to fit more closely to our intended retrieval scenario, the occurrence is not weighted by duration, meaning that a single instance of an erroneously recognised chord will add to the number of distinct chords per piece, regardless of the duration of that instance.

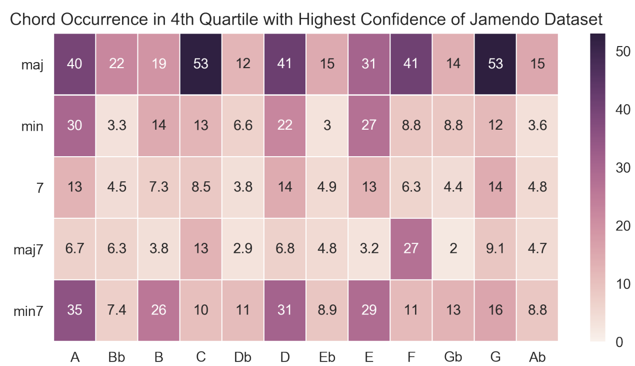
Therefore we drew the same statistics from the quarter of the dataset for which the confidence falls into the fourth quartile, the result of which is shown in Figure 4b. The average number of distinct chords per song and its standard deviation has significantly decreased to 9.04 ± 4.79, confirming our hypothesis of skew due to random error. We notice that the occurrence of the most popular chords Cmaj and Gmaj is nearly unchanged (53% instead of 52%), but the occurrence of the most popular tetrads has significantly decreased to a more expected level. The least popular chord is now Gbmaj7 and is only present in 2% of the files.

3.4 Architecture's Implementation

As shown in Figure 5, the web app prototype allows the user to query a set of chords from the Jamendo database by means of the *chord-db-as-a-service*. Next we detail the back-end and front-end modules of the prototype.



(a)



(b)

Figure 4: Percentage of files in the Jamendo dataset that contain the chords formed by the root in the columns and chord types in the rows.

3.4.1 Back-End. The chord sequence and confidence per file resulting from the offline analysis are stored in a MongoDB, a document-oriented type of database based on JSON. An additional dictionary is created per music piece. This contains, for each chord, the total proportion of the piece covered by that chord, which is used to facilitate the queries. That database is then deployed to the cloud such that it can be accessed from everywhere, providing a *chord-db-as-a-service.*

Upon receiving a set of chord symbols from the front-end, the database is queried using MongoDB's query language from the Python driver. Using the chord proportion dictionary, files get returned for which the specified set of chords covers the entire duration. In other words, only tracks that contain nothing but the specified chords get returned. This includes tracks that are entirely covered by a subset of the specified chords. So selecting a chord does not necessarily return a file that contains it, but not selecting a chord guarantees that it will not be in the files returned from the query. The reasoning behind this logic is that users can specify chords they know, and are guaranteed to only get music pieces that they are able to play along with. Therefore, the dataset does not get filtered by the query, but "gated", every additional chord allows more data to pass through.

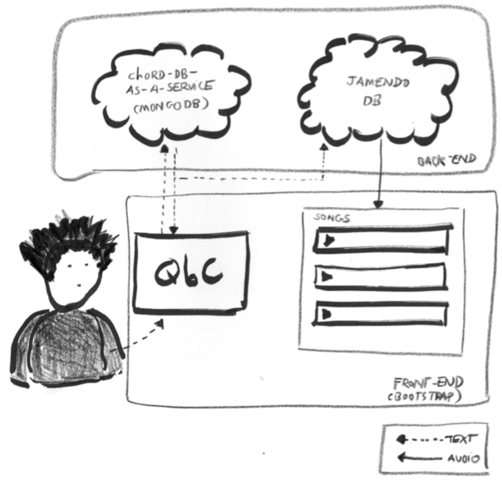


Figure 5: Architecture diagram of our QbC system.

For the small-scale pilot user study, three music pieces are selected from the query results and passed on to the front-end, according to two possible options. The first option selects the three pieces with the highest confidence, the second is a baseline that picks the first three results of the query, which we consider random.

3.4.2 Front-End. The front-end of the system has been implemented based on a Model-View-Control approach using Flask[11] for the model, Bootstrap[12] for the view and jQuery[13] and JavaScript for the controller. The model deals with the server side of the web app by managing queries sent to the *chord-db-as-a-service* database and formatting the results to JavaScript. The view shows two main spaces, the query space and the songs playlist space. In the QbC space, the user can pick a subset of chords by choosing chord types and notes shown as a tabular display. In the results space, a list of the songs with high confidence measure are provided. The controller adds the logic between the user's behavior and the communication with the server by transferring the user's query data. It also operates asynchronously with the Jamendo API[14] to serve the selected songs from QbC. A demo and additional information of the web app can be accessed online.[15]

4 USER'S EVALUATION

4.1 Study Design

Given that the app is web-based, we decided to conduct an online survey so that the users can do it in a real-world scenario: at their own environment and close to their musical instrument. As shown in Figure 6, participants should pick a subset of chords with types major, minor, 7th, major 7th or minor 7th. As shown in Figure 7, the study presents the participant with two lists of 3 songs each, corresponding to the two options in the back-end: a confidence

[11]http://flask.pocoo.org
[12]https://getbootstrap.com
[13]https://jquery.com
[14]https://developer.jamendo.com/v3.0
[15]https://www.audiocommons.org/jam

sorted list vs. a random sorted list. Retrieving 3 songs is a trade-off between showing only one result, which can be limiting, and showing a longer list that can be overwhelming. We are interested in finding out whether participants prefer one list over the other, given their chord selection. We are using a 7-point Likert scale to provide enough level of granularity [15] ranging from "strongly prefer list 1" (StP1), "prefer list 1" (P1), "slightly prefer list 1" (SlP1), "neutral" (N), "slightly prefer list 2" (SlP2), "prefer list 2" (P2) and "strongly prefer list 2" (StP2). We complement the rating with an input text box to enquire about the thought process behind their rating. Each user is asked to complete 5 independent trials, a fair number to complete in about 30 minutes. An added benefit is that if there is any problem with the completion of the overall survey, we can still use their partial response. The appearance of the lists for each trial are randomised.

The selection of 3 confidence-sorted or random pieces from the query results is also done independently, in order to make the comparison fair, but this also implies that there is overlap possible between the two lists. The chance of overlap will logically increase with a decreasing number of matches for the query, i.e. if rarer or less chords are selected. We are assuming that in the case of overlap between lists, the user will remain neutral with respect to the overlapping pieces and make their decision based on the non-overlapping pieces.

4.2 Task

We sent an invitation to the communities of music and music technology inviting predominantly novice practitioners to participate. We asked participants to pick a set of chords and rate and comment their preference from two given lists (confidence sorted vs. random sorted) of songs from the online database Jamendo. The only personal details required were aspects regarding their level of experience in music, their level of experience in music technology, the music instrument they were learning, gender and age. The entire experiment was designed to take 30 minutes or less to complete. Confidentiality of personal information and the anonymity of all participants is assured.

4.3 Iterative Design

At the time of writing, the prototype had been made available online and was accepting submissions. We plan to further develop the system according to an agile methodology (e.g. a follow-up study can be found in [14]) and continuously roll-out updates such that we can quickly observe their effect in practice. This decision is motivated by the fact that literature about query-by-chords is nearly non-existent, so there is no obvious starting point. Our first concern is to gather realistic chord queries, which we can then use to refine the matching procedure.

4.4 Participants

So far, we got responses from 7 participants (2 women and 5 men, with ages ranging from 26 to 64 years old), from now on P1–P7, resulting in a total of 35 independent trials. Seven users has been reported to be a sufficient number to obtain optimal results from a small-scale usability study [11]. This approach can help to report an early pilot study, where we identify existing problems of the prototype and refine the next design iteration, as discussed in this and the next section. The participants' musical skill level spanned from intermediate level (4 participants) and advanced level (3 participants). For the study, they used a keyboard (1 participant), a piano (3 participants), a guitar (1 participant), an acoustic guitar (1 participant) and a bass guitar (1 participant).

5 FINDINGS

Figure 8 shows the results as a 7-point Likert scale, which indicates that the most selected option was a slight preference for the confidence sorted list. The preferences for the random sorted list exhibit more variation. The difference between the two lists are not statistically significant however, according to Wilcoxon's signed-rank test [18].

We can get an idea about the initial chord queries by plotting them in a way similar to the chord occurrence in the entire database. The result is shown in Figure 9. It is already clear that users have a strong preference for triads, major chords and natural roots. It will be interesting to see if there are different trends to be observed according to instrument or skill level of the participants in a larger-scale pilot study that can provide more datapoints. It may also be beneficial to include novice musicians in the next evaluation phase and analyse whether there are any differences between beginners and more knowledgeable musicians. Overall, the most popular chords are the same as in the dataset but they appear considerably less frequent in queries. This can simply be explained by the fact that the average number of chords in a query is 2.94 ± 0.95, considerably less than the average number of distinct chords per file in the dataset.

An analysis of the participants' general comments from this preliminary study already gave us valuable feedback about how to improve the implemented algorithm and interface, demonstrating that our agile strategy is successful in that regard. An advanced level participant suggested to *"try to filter the collection to exclude jingles (the majority of results), songs with no chords (hopefully the algorithm should include that) and ideally focus on songs with clear chord sequences"* (P1). Two participants suggested to avoid *"ambient stuff with many notes and very ambiguous harmony"* (P1) and *"drones"* (P7). The fact that the two lists could include two repeated tracks and that there were also tracks that were repeated from previous rounds was unclear to the participants: *"I would like to hear new tracks between rounds"* (P2). An intermediate level participant suggested to include the scores to make the task less difficult: *"Even though I can play some complex pieces, I've found the exercises rather difficult"* (P5). One participant proposed to *"have the possibility to choose the musical styles"* (P5).

6 LESSONS LEARNED

The results from this small-scale pilot study indicate that the interface design and implemented algorithm are in the right direction but there is still room for improvement. We found out that having a prototype running is useful to continuously getting data that can inform how to (1) refine the algorithm from understanding about optimal queries and (2) refine the interface from user's behaviour. As per the time of writing this paper, although we have recruited

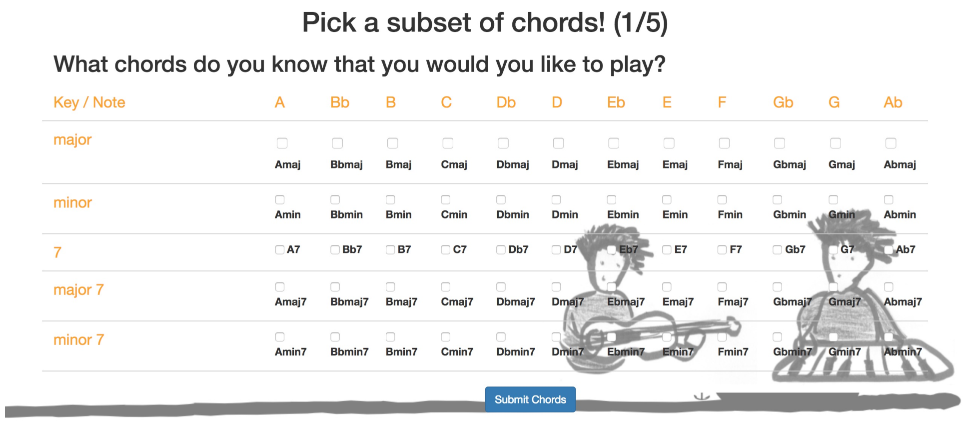


Figure 6: Screenshot of the query-by-chords functionality in *Jam with Jamendo*.

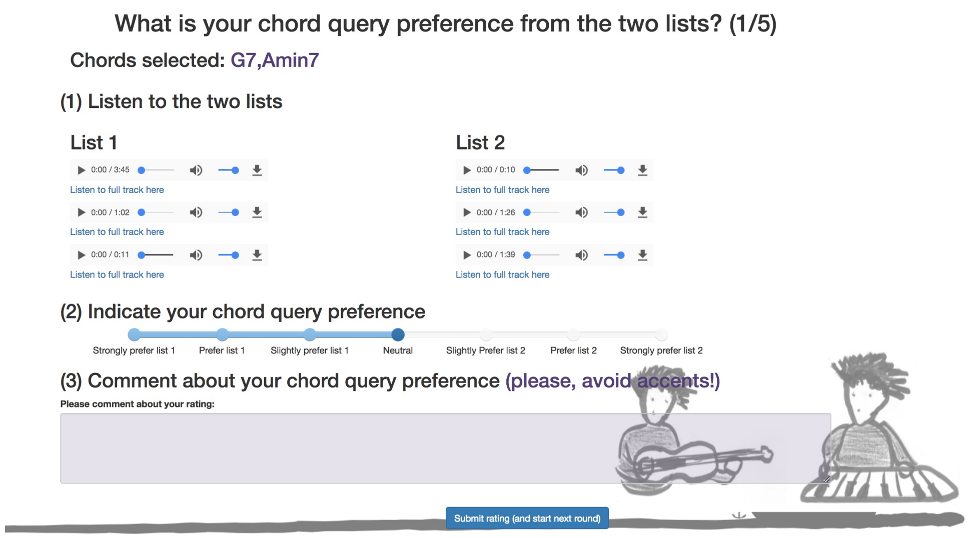


Figure 7: Screenshot of the evaluation of a trial with two results: confidence ordered list vs. random ordered list.

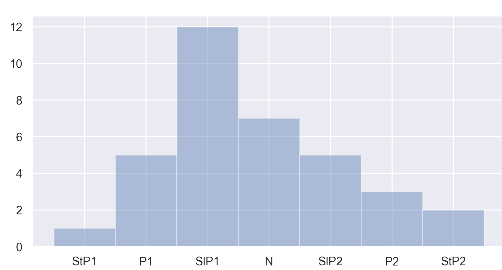


Figure 8: Bar plot of the ratings of the two lists, where list1 is equal to the confidence sorted list (N = 35).

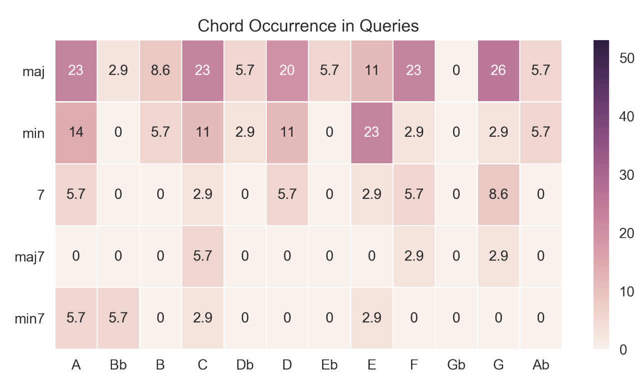


Figure 9: Chord occurrence in sample queries.

a small number of participants for conclusive results, we have noticed that users need to be nudged towards longer queries and that we should incorporate visual feedback with information about the sequence of chords in the songs. Apart from music experts who are able to recognise the presence of chords by ear, there should be a mechanism by which music beginners could validate whether songs from the lists do contain the queried chord. The lack of such a validation mechanism makes users focus on the selection of the songs for their ratings. In order to evaluate the effectiveness of the confidence measure, some sort of visual feedback needs to be implemented first and then an experiment could be created where users are explicitly instructed to compare the quality of the chord transcriptions between the two lists.

Sorting by confidence measure has the unintended effect that the same music pieces keep on getting returned as optimal matches to the queries, which leads participants to prefer a random list because of variety. A certain degree of randomness will therefore be encouraged in future iterations of the app. Furthermore, the combination of the current query matching mechanism and the confidence sorting has the effect that short pieces get promoted, because it's statistically less likely that transcription errors are made in them. A weighting based on duration could be applied to alleviate this, or a more complex query matching such as the one detailled in [12] could be implemented.

Given that the distribution of chords is data specific or to aid beginners, it would be interesting to highlight the chords that tend to occur together when selecting chords, so that the reliability of getting results is assured. This approach promotes the exploration of common combinations and it would increase the average number of matching music pieces per query. However, special care should be taken to not obstruct users who deliberately want to explore uncommon combinations.

7 CONCLUSION

In this paper, we presented the web app prototype *Jam with Jamendo*, which promotes the exploration of music from the online music database Jamendo based on query-by-chords and a computational confidence measure. The prototype is especially designed for learners of an instrument who want to practice the chords they know and discover new digital music. We presented a small-scale pilot study focusing on the query task to assess the prototype and algorithm implementation. We found out that we are moving in the right direction and that having the prototype as an experimental playground is helpful to refine the algorithm and interface for the music learner's needs.

As future work, we plan to improve the visual feedback of the songs with their chord sequences so that it is useful in a music education context. We aim to explore other datasets as well (e.g. YouTube) and create meaningful graphical representations that are instrument-agnostic. We are also interested in proposing a more dynamic and customizable approach to the confidence measure so that participants can choose flexibly between getting more diverse, but possibly lower quality results, or less results, but of higher quality.

8 ACKNOWLEDGEMENTS

We are grateful to the participants of the study. Thanks to Miguel Ceriani for his advice on the database back-end implementation. AudioCommons is funded by the European Commission through the Horizon 2020 programme, research and innovation grant 688382. This work has been partly funded by the UK Engineering and Physical Sciences Research Council (EPSRC) grant EP/L019981/1.

REFERENCES

[1] M. Barthet, A. Anglade, G. Fazekas, S. Kolozali, and R. Macrae. 2011. Music Recommendation for Music Learning: Hotttabs, a Multimedia Guitar Tutor. In *Workshop on Music Recommendation and Discovery*. 7–13.

[2] M. Barthet, M. D. Plumbley, A. Kachkaev, J. Dykes, D. Wolff, and T. Weyde. 2014. Big Chord Data Extraction and Mining. In *Proceedings of the 9th Conference on Interdisciplinary Musicology*.

[3] M. A. Casey, R. Veltkamp, M. Goto, M. Leman, C. Rhodes, and M. Slaney. 2008. Content-based Music Information Retrieval: Current Directions and Future Challenges. *Proceedings of the IEEE 96* 96, 4 (2008), 668–696.

[4] B. de Haas, J. P. Magalhaes, D. Ten Heggeler, G. Bekenkamp, and T. Ruizendaal. 2014. Chordify: Chord Transcription for the Masses. In *Proceedings of the 13th International Conference on Music Information Retrieval Late Breaking and Demo Session*.

[5] F. Font, T. Brookes, G. Fazekas, M. Guerber, A. La Burthe, D. Plans, M. D. Plumbley, M. Shaashua, W. Wang, and X. Serra. 2016. Audio Commons: Bringing Creative Commons Audio Content to the Creative Industries. In *Proceedings of the 61st AES International Conference: Audio for Games*. Audio Engineering Society.

[6] F. Font and X. Serra. 2016. Tempo Estimation for Music Loops and a Simple Confidence Measure. In *Proceedings of the 17th International Conference on Music Information Retrieval*. 269–275.

[7] J-S. R. Jang, H-R. Lee, and J-C. Chen. 2001. Super MBox: An Efficient/Effective Content-Based Music Retrieval System. In *Proceedings of the Ninth ACM International Conference on Multimedia*. 636–637.

[8] P. Knees, T. Pohle, M. Schedl, and G. Widmer. 2007. A Music Search Engine Built upon Audio-based and Web-based Similarity Measures. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. 447–454.

[9] L. Lessig. 2004. *Free Culture: How Big Media Uses Technology and the Law to Lock Down Culture and Control Creativity*. The Penguin Press.

[10] M. Mauch, C. Cannam, M. Davies, S. Dixon, C. Harte, S. Kolozali, D. Tidhar, and M. Sandler. 2009. OMRAS2 Metadata Project 2009. In *Proceedings of the 10th International Conference on Music Information Retrieval*.

[11] Jakob Nielsen and Thomas K. Landauer. 1993. A Mathematical Model of the Finding of Usability Problems. In *Proceedings of the INTERACT '93 and CHI '93 Conference on Human Factors in Computing Systems*. 206–213.

[12] J. Pauwels, G. Fazekas, and M. B. Sandler. 2018. Recommending Songs to Music Learners Based on Chord Content. In *Proceedings of the 2018 Joint Workshop on Machine Learning for Music*.

[13] J. Pauwels, K. O. Johan, G. Fazekas, and M. B. Sandler. 2017. Confidence Measures and Their Applications in Music Labelling Systems Based on Hidden Markov Models. In *Proceedings of the 18th International Conference on Music Information Retrieval*. 279–285.

[14] J. Pauwels, A. Xambó, G. Roma, M. Barthet, and G. Fazekas. 2018. Exploring Real-time Visualisations to Support Chord Learning with a Large Music Collection. In *Proceedings of the Web Audio Conference*.

[15] C. C. Preston and A. M. Colman. 2000. Optimal Number of Response Categories in Rating Scales: Reliability, Validity, Discriminating Power, and Respondent Preferences. *Acta Psychologica* 104, 1 (2000), 1–15.

[16] G. Tzanetakis and P. Cook. 2000. 3D Graphics Tools for Sound Collections. In *Proceedings of the Conference on Digital Audio Effects*.

[17] C. West. 2013. Motivating Music Students: A Review of the Literature. *Update: Applications of Research in Music Education* 31, 2 (2013), 11–19.

[18] F. Wilcoxon. 1945. Individual Comparisons by Ranking Methods. *Biometrics Bulletin* 1, 6 (December 1945), 80–83.

[19] E. Wold, T. Blum, D. Keislar, and J. Wheaten. 1996. Content-based Classification, Search, and Retrieval of Audio. *IEEE Multimedia* 3, 3 (1996), 27–36.