PATTERNS UI, AN INTERACTIVE TOOL FOR MUSIC EXPLORATION

Rory SWEENEY (rory.sweeney@universityofgalway.ie) (0009-0005-5439-2032)[1],
Pushkar JAJORIA (pushkarjajoria@gmail.com) (0009-0006-3789-5372)[1],
Danny DIAMOND (d.diamond1@universityofgalway.ie)[1],
Mathieu D'AQUIN (mathieu.daquin@loria.fr) (0000-0001-7276-4702)[2], and
James MCDERMOTT (james.mcdermott@universityofgalway.ie) (0000-0002-1402-6995)[1]

[1]**Data Science Institute and School of Computer Science, University of Galway**, Galway, Ireland
[2]**Laboratoire Lorrain de Recherche en Informatique et Applications and Institut des Sciences du Digital, Université de Lorraine**, Nancy, France

ABSTRACT

Patterns UI is a novel user interface designed for the exploration of trends within symbolic music corpora through shared melodic patterns. It is an open source tool that caters to musicologists, musicians and enthusiasts, enabling users to discover, compare, and visualize melodic patterns in European folk music. These patterns are represented as short, integer n-grams of accented notes, based on previous musicological research. A central feature is an interactive network visualisation showing tunes, patterns, and the relationships among them, as well as tune families. The interface also features a fuzzy search for tune titles, a search by pattern, and a metadata search. Patterns UI was co-designed and then evaluated with potential users, including musicologists.

1. INTRODUCTION

Patterns UI is a simple, modular piece of software for exploring a collection of tunes and their shared melodic patterns, created using previously-developed pattern detection algorithms, stored as a knowledge graph (KG). In particular it allows direct exploration of the graph as nodes and edges, with two different views, and for various types of search. A main view is shown in Figure 1.

The goal of the work presented was to develop an interactive tool for exploring and visualising the information contained in the KG to satisfy the needs of musicologists, composers, musicians and enthusiasts. As these users might not possess the proficiency to derive meaningful insights from KGs through direct SPARQL queries against the KG, our objective was to design an interface facilitating users to search and explore within the KG, taking advantage of visualisation, search, information hiding, and more.

The data to be explored in the Patterns UI is a database of melodic patterns extracted from Irish folk tunes. The tune corpus is *The Session*, a central resource in the Irish folk community [1]. The familiar n-gram method is used to extract all relevant patterns from a subset of the corpus. An n-gram in our context is a sequence of consecutive notes of length n in a tune. We also extract a measure of their importance in the tune, based on TF-IDF. TF-IDF, or *Term Frequency–Inverse Document Frequency*, is a method of determining the importance of a term, or in our case an n-gram, to a document by comparing its frequency in the document to its occurrences across an entire corpus. An n-gram with a high TF-IDF value is common in the tune but not common in the overall corpus. Using this method, the melodic patterns most characteristic of each tune can be identified. Finally we extract a complexity measure, based simply on the number of unique notes in the n-gram pattern. The patterns are stored in a knowledge graph (KG) together with metadata on tunes.

Since a tune contains multiple patterns, and the same pattern can be contained in multiple tunes, our data forms a network. Moreover, we have metadata on *Tune Family* membership for a small subset of the corpus, which can be explored and visualised in the network. We will describe the Irish traditional music context, the pattern extraction, metadata, and tune family data in more detail in Section 2. This section also includes information on the processing of pattern data to produce a KG.

Our focus is then on how to explore such a KG. Starting from competency questions (CQs) we have drawn inspiration from previous UIs for exploration of musical corpora and of networks in general, described in Section 3. We engaged in a co-design process with potential users such as computational musicologists. We describe this process in Section 4, leading to a statement of requirements.

We describe the app itself in Section 5, with discussion and evaluation of strengths and limitations. Notes on evaluation are in Section 6. We offer conclusions and future work in Section 7.

2. BACKGROUND

2.1 Patterns and Tune Families in Irish Folk Music

The existence of common melodic patterns[1] between tunes in the Irish tradition was first noted by 19th century collectors George Petrie and William Forde [3]. The influential Irish-American music collector Francis O'Neill de-

[1] The following section draws on the M.Sc. thesis of author Danny Diamond [2].

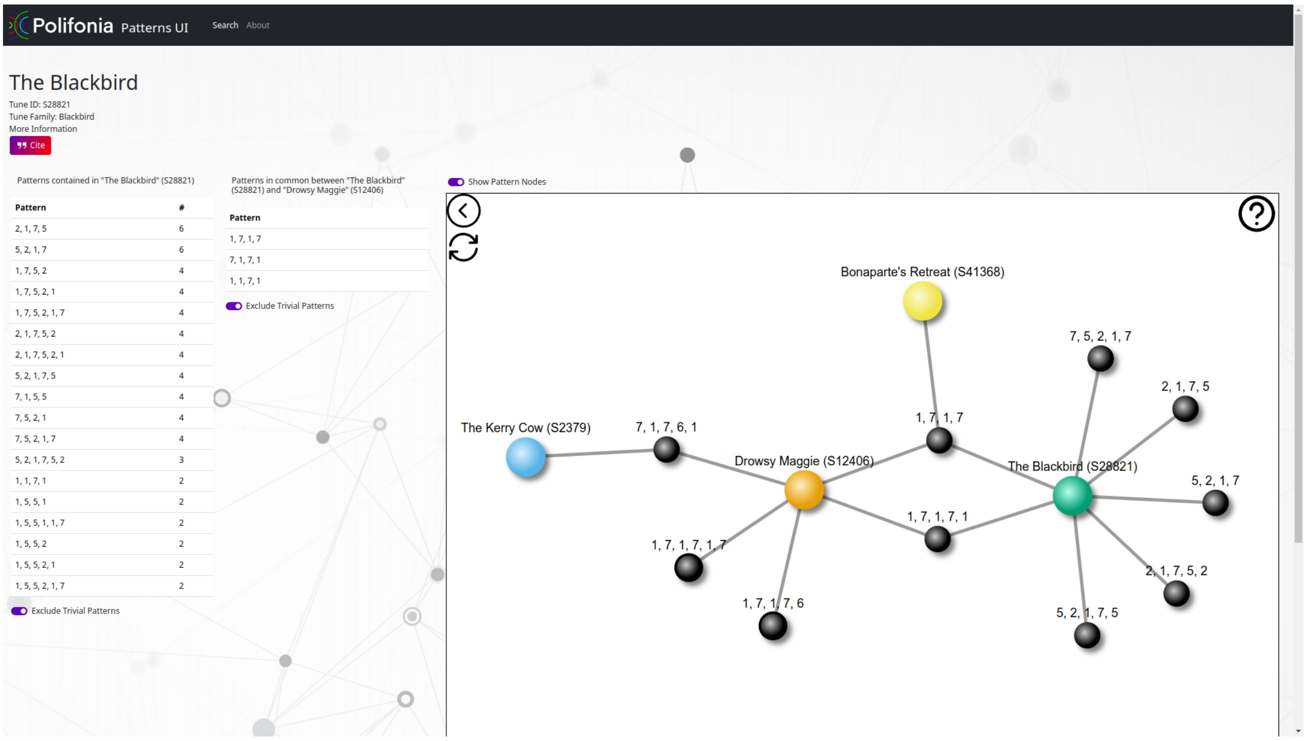

Figure 1. Patterns UI *Composition* page. On the left, a table of the most common patterns in the selected tune; a middle panel containing a list of patterns in common with the previously selected tune; on the right, a network visualisation showing the current tune, some of its most common patterns (small, black nodes), and other tunes that contain the same pattern. At the top of the page, the tune title and tune family name are displayed above a link to the original source for the tune and a *Cite* button.

voted a chapter in his 1913 book Irish Minstrels and Musicians to the qualitative identification of related melodies within the Irish tradition [4].

In 1950, American folk music collector and researcher Samuel Bayard published his theory of *tune families*: proposing that many traditional melodies could be traced, in the manner of a family-tree, to prototype melodies [5]. In Bayard's words:

"*...a group of melodies showing basic interrelation by means of constant melodic correspondence, and presumably owing their mutual likeness to descent from a single air that has assumed multiple forms through processes of variation, imitation and assimilation.*"

Bayard's tune family concept has been widely adopted in research on Irish and other Western folk music [3, 6, 7].

This provides the point of departure for our work. We assume that these important shared patterns can be captured well by a diatonic representation. Indeed, the diatonic representation deals well with the modal ambiguity common in Irish traditional music. We deal with transpositions among tune variants by normalising all melodies to a common tonal centre.

Further, we take advantage of previous work on structure in Irish traditional music to focus on "accented tones" only, i.e. those on the beat. Their importance (in contrast to the variability of the unaccented tones) in defining a stable outline melody crops up consistently in the literature [6, 8–10].

Thus, while we have worked with multiple definitions of patterns, in the current work we represent a pattern as an n-gram of integers representing a sequence of diatonic scale degrees of accented notes, which occurs more than once in the corpus. For example, the pattern (7, 1, 7, 6, 1) is a 5-gram pattern visible in Fig. 1.

The Irish folk music corpus "The Session" [2] (approximately 40,000 tunes) was used as a source of tune data in ABC notation, and converted to MIDI and then to the integer representation. For 314 tunes, we have annotated them with ground-truth on tune family membership. A further source is the *Essen* corpus [3], with tune data available through the MTCFeatures library [4].

2.2 Linked Data

Throughout the Polifonia project, we have represented datasets as linked data (ontologies and KGs). The motivation for this approach was to enable flexible and efficient querying, open access, preservation, and integration across multiple data sources [11]. Using well-designed ontologies and KGs allows current data to integrate well with future data also.

So, the patterns we have extracted are stored in a Patterns KG, which itself takes advantage of ("lives on top of") an additional KG representing tune metadata [5]. The construction of these KGs is described in previous work [11],

[2] https://thesession.org/
[3] http://essen.themefinder.org/
[4] https://github.com/pvankranenburg/MTCFeatures
[5] https://github.com/polifonia-project/tunes-knowledge-graph

and draws on previous work in ontologies and KG creation within the Polifonia project [12–14].

The *Patterns2KG* pipeline was used to extract patterns from the tune data and process them into a KG in RDF format, together with tune family membership data. Information is obtained from the KG using the SPARQL query language [11], typically via a SPARQL endpoint.

3. PREVIOUS WORK

Tovstogan et al. used a user interface for the exploration and discovery of personal music collections in which labelled segments of musical compositions are represented as points in a 2D visualisation. Dimension reduction algorithms were used to position points in the visualisations and suggest similarity. They found that users perceived this system as engaging, rewarding and useful. The authors suggest that similar systems could be extended to larger music collections [15].

Knees et al. developed a three-dimensional interface for exploration of musical collections, in which similar compositions are clustered together, and their properties processed into a height value to form a landscape through which the user can navigate a music collection. A user study of 8 people received positive responses [16].

The *Digital Music Lab* system of Abdallah et al. employs semantic web technologies to support large-scale musicological research across large music corpora. Their system integrates into Linked Open Data, allowing for distributed system across multiple corpora. It includes an interactive graphical interface through which analyses of musical collections can be visualised as bar charts, line graphs or histograms. Their user interface does not, however, include a network visualisation. [17].

Walshaw developed the abcnotation.com web interface to search an online corpus of traditional music stored in ABC notation. The interface provides tools to view tunes in musical notation and play MIDI audio. The interface also features the *TuneGraph* network visualisation, which presents tunes as nodes with connections to a small set of similar tunes. Nodes can be double-clicked to navigate to the selected node's page, with a new small network diagram generated for that tune. The *abcnotation.com* interface is depicted in Figure 2 [18–20].

The *TunePal* interface, `https://tunepal.org`, allows querying a corpus of traditional Irish, Welsh, Scottish and Breton music by title and by playing music. The score of the selected composition is displayed on a musical staff and in ABC notation. MIDI and recorded audio of tunes can be played. There are no visualisations on the *TunePal* webpage, and the target audience is musicians rather than musicologists. [21–23]

As part of the Polifonia project, a harmonic relation graph was produced, showing inferred relationships between pairs of pieces of music (see Figure 3). This was presented at the *Sonar Festival* Music and AI event in October 2021. In this graph, each edge represents an inferred relationship between a pair of pieces of music. This graph is static, but uses graph layout algorithms to produce a strong sense of overall geometry of the implicit 2D musical space.

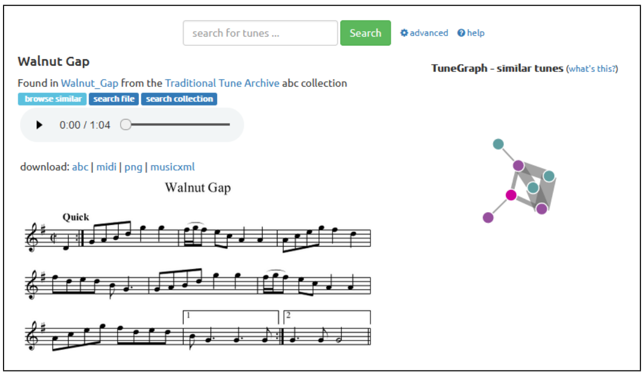

Figure 2. The *Abcnotation* interface, e.g. `https://abcnotation.com/tunePage?a=tunearch.org/wiki/Walnut_Gap.no-ext/0001` demonstrates the visualisation of a small network of similar tunes.

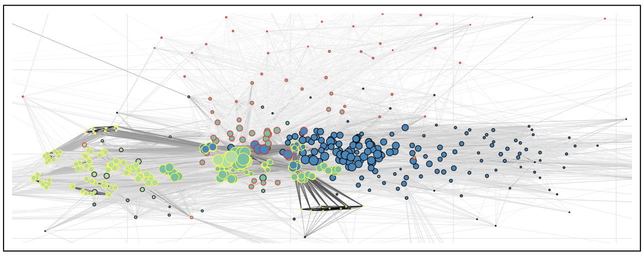

Figure 3. A harmonic relation graph, part of the Polifonia INTERLINK pilot. Figure from [24].

Finally, we mention the *LOD Live* interface, designed for exploration of arbitrary KGs, with an example shown in Figure 4. This UI is not music-oriented, but served as inspiration for a live network diagram with show/hide functionality based on linked open data.

This review of pre-existing interfaces produced a list of desirable interface features including search facilities over metadata and over contents, display of patterns on musical staves, audio playback of patterns and a network diagram for exploration of relationships.

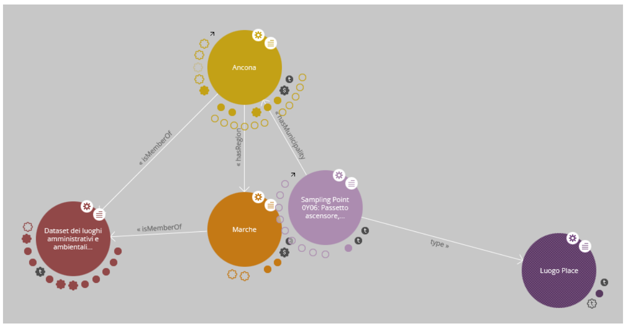

Figure 4. The *LOD Live* interface, e.g. `http://en.lodlive.it/?https://w3id.org/italia/env/ld/place/municipality/00201_042002`.

4. DESIGN PROCESS

This section describes how we went about designing the interface and software, incorporating potential users' feedback at early stages.

4.1 Wireframe Design

The initial phase of our project involved the task of designing the website's wireframe. This step helped to decide the overall structure of the website, ensuring that it was both user-friendly and tailored to the specific needs of our target audience – musicologists. The design phase also involved planning of key features, such as a search interface that borrowed elements from previous implementations while remaining flexible to accommodate various user input formats. Wireframe designs for the *Search* interface, *Composition* page and *Pattern* page were particularly used during our co-design process.

4.2 Co-Design Process

A significant aspect of our Co-Design process was conducting remote, semi-structured interviews with musicologists. Initial wireframes and/or related interfaces were presented to each participant, followed by an open discussion with brainstorming about possible features. Sessions lasted 20–50 minutes. Six such interviews were conducted with musicologists from varying backgrounds.

The insights provided by these discussions into the practical applications and user expectations of our website were instrumental in designing our user interface. The proposal of a network diagram as a method of interaction was received particularly positively, underlining its potential as a musicological tool. The minutes and feedback from these meetings were archived and guided the work that followed.

Key takeaways included the need for a search interface leading to a list of results, the importance of visual cues like hovering and highlighting for information presentation, and the necessity of a flexible network diagram to explore and navigate the large corpus. Furthermore, these interactions emphasized the importance of catering to both casual and expert users, balancing immediate usability with the depth of information for more experienced musicologists.

During the design phase, the need for citeability of particular findings was also raised by project data management researchers.

4.3 Stories and Competency Questions

Within the Polifonia project, resulting from stakeholder interviews, personas and competency questions (CQs) were created to guide design processes within the project. We drew on these during design and development [6]. The ones most relevant to our work are listed below.

For the persona *Brendan*, who is a traditional musician and a lecturer in ethnomusicology and Irish traditional music, we sought to address the following CQs.

- What tunes are similar to tune X, given similarity measure Y?
- What are the metadata for collection X?

Persona *Mark* is a computational musicologist, researcher and lecturer. The following CQs were created to address his needs from the system.

- What are all compositions in tune family X?
- What are the similarities / differences of all compositions in tune family X according to measure Y?

The persona *Sophia* is a musicologist and a practising musician, for whom the following CQ was considered.

- What different motifs exist in a piece of music?

4.4 Requirements

Based on the above, we arrived at the following list of requirements:

- An *About* page with information, links, and explanations.
- A *Search* function, allowing search by different fields, including text fields like 'title', and structured fields like 'key signature'.
- Fuzzy *Title* search to help deal with variant spellings of titles.
- Search by pattern, allowing different methods of representing a pattern (e.g., space-separated integers versus comma-separated).
- A *Composition* page, showing details of a particular tune, relevant patterns, and external links to additional information.
- A *Pattern* page, to show the details and relevant tunes for a particular pattern, and visualise a pattern on a musical staff, with the ability to listen to it directly.
- A network diagram, showing tunes and patterns as nodes, with links representing a pattern present in a tune.
- A feature to indicate tune family membership.
- A feature to allow citation of any finding, e.g. by citing a specific URL.

4.5 Technology Stack

We created a modular design. The KGs, developed previously, are hosted in a Blazegraph instance via a Polifonia-custom Flask app providing a SPARQL endpoint [7].

The backend server was written using the Python Flask web framework, and communicates with the KG using the SPARQL query language. The Vue.js JavaScript framework was used for front-end development.

The source code repositories for both the frontend [8] and backend [9] are open and accessible on GitHub.

[6] https://github.com/polifonia-project/stories

[7] https://polifonia.disi.unibo.it/fonn/sparql

[8] https://github.com/polifonia-project/pattern-exploration-GUI

[9] https://github.com/polifonia-project/pattern-explorations-backend

5. USER INTERFACE

In this section, the final design of the application is described from the point of view of the user. The interface opens onto a *Search* page which features three search methods for finding tunes.

The first search method is a tune *Title* search. The fuzziness of its matching algorithm allows it to accept partial titles or misspellings as search terms and still return relevant results. The search results show the tune name and tune ID for each result, as well as relevant metadata such as tune type (e.g. jig, reel, strathspey), key and time signature, as shown in Figure 5. Clicking on a search result opens the *Composition* page for the selected tune.

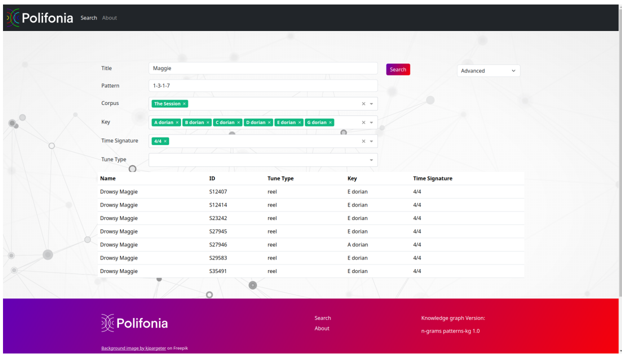

Figure 5. The *Search* page showing the *Advanced* search interface. The *Title* field contains the partial title 'Maggie', which can match a number of tune titles as a result of the fuzzy *Title* search feature. The pattern '1, 3, 1, 7' has been entered in the *Pattern* field using hyphens as a delimiter. 'The Session' has been selected from the *Corpus* drop-down. The *Key* drop-down further limits the search to tunes featuring a Dorian key, and the *Time Signature* drop-down specifies a '4/4' time signature. The results of the search are shown in a table.

The second type of search is a *Pattern* search, which finds tunes containing a melodic pattern inputted as a sequence of delimited numeric scale degrees.

Finally, the *Advanced* search incorporates the fuzzy *Title* and *Pattern* search fields, along with multiple-select drop-down boxes for 'Corpus', 'Key', 'Time Signature', and 'Tune Type', the options of which can be filtered using associated text fields.

An example of a *Composition* page is shown in Figure 1. On the *Composition* page, the most prominent feature is a network diagram. This represents tunes as large nodes, coloured according to the tune's family. The slightly smaller *Pattern* nodes, in black, represent melodic patterns. A connection between a *Tune* node and a pattern node indicates that that pattern is contained in the tune in question.

The network visualisation is interactive and can be expanded by clicking on nodes. A *Tune* node can be clicked to reveal additional patterns contained in the selected tune, and a *Pattern* node can be clicked to reveal additional tunes containing that pattern. Double-clicking a tune node navigates to the clicked node's *Composition* page, while double-clicking a *Pattern* node navigates to this node's *Pattern* page. If there are no additional connected nodes to expand when a node is clicked, the perimeter of the clicked node flashes red to indicate this.

Nodes in the network diagram can be rearranged by dragging with the mouse pointer, and the diagram can be zoomed by scrolling. Two icons at the top left of the diagram allow for undo and reset functions for the visualisation. A *Help* icon in the top-right corner of the visualisation, when clicked, displays a short series of tutorial panes that explain how to use the network visualisation. An example of one of these tutorial panes is shown in Figure 6.

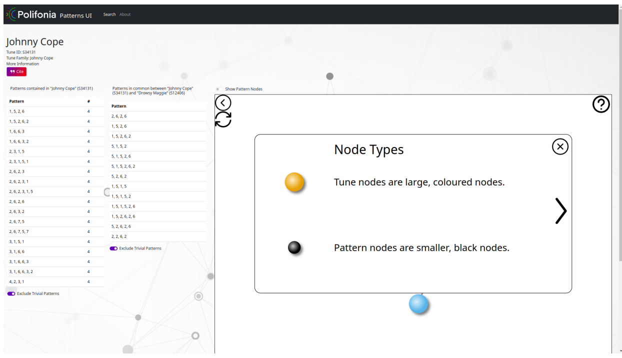

Figure 6. The network visualisation tutorial pane is opened by clicking on the *Help* icon. The tutorial panel shown lists the types of nodes used in the visualisations.

A toggle switch above the network visualisation allows the pattern nodes to be hidden, resulting in a network visualisation containing only tune nodes connected to other tune nodes based on shared patterns. This configuration is shown in Figure 7.

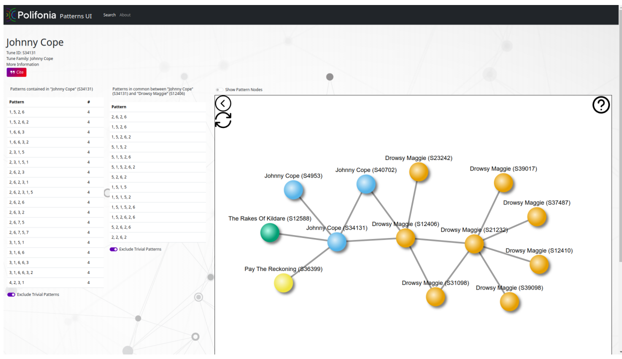

Figure 7. The *Composition* page featuring the network diagram with pattern nodes disabled. This network diagram shows the selected tune and similar tunes based on the numbers of occurrences of shared patterns weighted by pattern complexity. Contrast this with Figure 9 which shows the same tunes, but also shows the patterns.

Additionally, the *Composition* page features a panel on the left side listing the most common melodic patterns contained in the selected tune along with their occurrence frequencies. Trivial patterns can be excluded from the list by adjusting a toggle switch.

Trivial patterns are patterns with low pattern complexity. Pattern complexity is a measure of the importance of a pattern in a tune, defined as the proportion of unique note values in the pattern. For example, the pattern "5, 5, 5, 5" is very common in the corpus. However, since it is repetitive, its pattern complexity is low, and so it is likely not a distinctive melody associated with that tune. It is de-emphasised as a result of its low complexity.

When a tune node is double-clicked, the selected tune's *Composition* page will load and feature a middle panel which lists the melodic patterns in common between this tune and the tune described by the previous *Composition* page. Here too, trivial patterns can be excluded by adjusting a toggle switch. The patterns listed in both panels can be clicked to navigate to their respective *Pattern* pages.

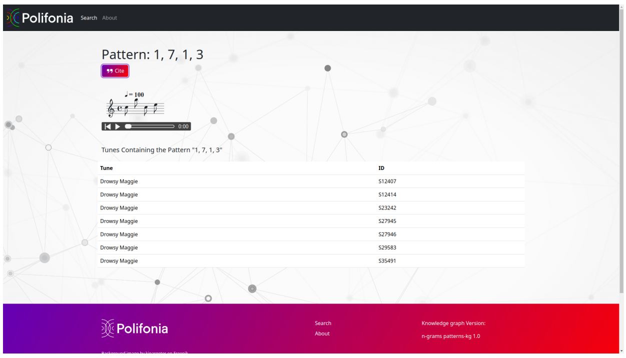

Figure 8. The *Pattern* page shows a musical stave featuring the pattern in musical notation, as well as a MIDI player which can play audio of the pattern. Below them is the full list of tunes containing the selected pattern. Below the page title, is the *Cite* button.

The *Pattern* page features a musical stave representing the melodic pattern described by the page in musical notation, along with a MIDI player that can play audio of the pattern. Below the stave, there is a list of the tunes that contain the selected pattern, each entry linking to its respective *Tune* page. A screenshot of the *Pattern* page can be seen in Figure 8.

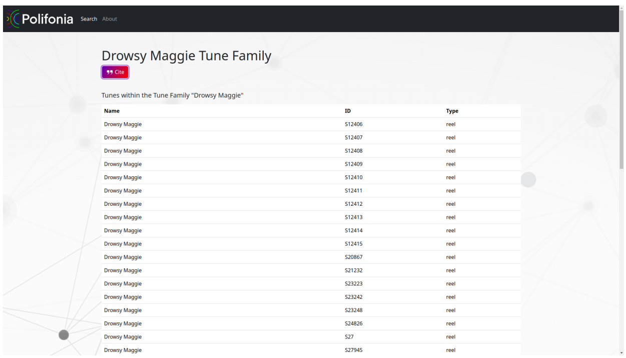

Figure 9. The *Tune Family* page shows a list of tunes contained in the selected tune family. Below the tune family page title, there is the *Cite* button.

The *Tune Family* page for a given tune can be accessed from a link on its *Composition* page below the page title. The *Tune Family* page lists all the tunes in the selected tune family. Each tune in the list links to its respective *Composition* page. An example of a *Tune Family* page is shown in Figure 9. The *Composition* page also features a link to external resources for each tune.

The *About* page provides information about the interface, including links to the project GitHub repositories, information about the Polifonia project, the funding associated with the project and a link to a page providing the Patterns knowledge graph endpoint and example SPARQL queries that can be used with it.

On several pages, there is a feature that generates a citation for the *current* page, which can be copied to the clipboard using the *Copy* button. This provides easy citability for specific findings, such as a specific pattern's presence in a specific tune.

The footer on each page displays the current *Patterns* knowledge graph version string as well as links to the *Polifona Web Portal* and Polifonia social media pages.

6. EVALUATION

6.1 Usability Testing

Usability testing evaluations were conducted with 2 participants (not part of the project) during February 2024.

- Subject 1 is a PhD student, working on musical complexity and patterns, and music generation, working on both Irish and Iranian melodic music.
- Subject 2 is an experienced music technology researcher, working as a lecturer.

For each evaluation, the subject was given a very brief explanation of the UI and KGs - tunes, patterns, links between them, and searches. The subject was then given several tasks to be carried out, while two observers watched, made notes, answered Subject's questions, and gave hints, only when needed. Sessions lasted approximately 30 minutes. The tasks given to the subjects were derived from our co-design process, competency questions, and requirements, as described in previous sections.

Subjects were generally successful in tasks involving searches. The fuzziness of the title search proved useful for unfamiliar song names, and participants had no difficulty with pattern searches. For advanced searches, Subject 1 did not initially realise that multiple drop-down options could be selected, but both participants eventually completed the task successfully.

The purposes of the pattern panes were quite clear to both participants, though the appearance of the middle pane, for patterns in common with the previous tune, was not initially noticed by Subject 2.

Some features of the network visualisation were confusing for the participants. The existence of the hover feature to reveal the tune family name was not initially apparent to Subject 1, and the existence of distinct single and double click features was not clear. The red ring flash visual feedback for the case when no additional neighbour nodes can be expanded was not noticed by one participant.

Despite this, both participants were able to successfully achieve tasks related to the network visualisation such as finding additional tunes containing a given pattern and finding patterns in common between two tunes.

Some of the links on the composition page and in the page footer were found not to be sufficiently prominent and not found by the participants.

Using the *Pattern* page to play MIDI audio of patterns and using the *Tune Family* page to find tunes in the same family as the selected tune were easily accomplished by both participants.

6.2 Walkthrough Demonstration and Evaluation

An open walkthrough and demonstration of the interface was held with a large group of project partners during February 2024. The participants included academics in the fields of Computer Science, Human-Computer Interaction and Musicology. This involved a ten-minute, remote demonstration of all features of the application, in combination with the option of participants using the interface directly through a provided link to the live website. The most important feedback here was to improve user-friendliness for first-time users by providing better visual differentiation between elements of the network diagrams, and pop-up help; and to provide code resources such as SPARQL queries on a Resources page.

All feedback was acted on for the current version of the UI described in the previous section.

7. CONCLUSIONS AND FUTURE WORK

In this paper we have presented a novel user interface, Patterns UI, for exploration of a collection of tunes and patterns. Usability testing produced positive reactions, demonstrated the interface's usability, and produced useful feedback. The UI clearly answers the competency questions.

Planned future work on the system includes offering multiple tune similarity metrics with which to link tunes in the tune-tune network visualisation, such as alignment similarity and similar methods developed by Van Kranenburg and colleagues [25, 26].

In contrast to our melody-focussed work, other methods of measuring relatedness between pairs of pieces of music could also be used. We plan to use the UI for interactive visualisation of relationships among pieces of music based on chord- and harmony-based knowledge graphs, such as the *ChoCo* or *Harmory* KGs [27–29].

Acknowledgments

Patterns UI, as part of the Polifonia Project, has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement N. 101004746. This publication has emanated from research supported in part by a grant from Science Foundation Ireland under Grant number [12/RC/2289_P2] the Insight SFI Research Centre for Data Analytics. For the purpose of Open Access, the author has applied a CC BY public copyright licence to any Author Accepted Manuscript version arising from this submission.

8. REFERENCES

[1] (2024) The Session. Accessed 2024-02-07. [Online]. Available: https://thesession.org/

[2] D. Diamond, "Automatic tune family detection in a corpus of Irish traditional dance tunes," Master's thesis, University of Galway, 2024, forthcoming.

[3] J. R. Cowdery, "A fresh look at the concept of tune family," *Ethnomusicology*, vol. 28, no. 3, p. 495, Sep. 1984. [Online]. Available: https://www.jstor.org/stable/851236?origin=crossref

[4] F. O'Neill, *Irish minstrels and musicians*. Chicago, USA: Regan Printing House, 1913.

[5] S. P. Bayard, "Prolegomena to a study of the principal melodic families of British-American folk song," *The Journal of American Folklore*, vol. 63, no. 247, pp. 1–44, 1950, publisher: University of Illinois Press. [Online]. Available: http://www.jstor.org/stable/537347

[6] M. Ó Suilleabháin, "The creative process in Irish traditional music," in *Irish musical studies 1: musicology in Ireland*, ser. Irish musical studies. Dublin, Ireland: Irish Academic Press, 1990, vol. 1, pp. 117–130.

[7] A. Hillhouse, "Tradition and innovation in Irish instrumental folk music," Master's thesis, The University of British Columbia,, Vancouver, Canada, 2005.

[8] B. Breathnach, "Between the jigs and the reels," 1982. [Online]. Available: http://msikio.online.fr/Breathnach/breandn.htm

[9] D. Ó Maidín, "A programmer's environment for music analysis," Ph.D. dissertation, University of Limerick, Limerick, Ireland, 1995. [Online]. Available: http://rgdoi.net/10.13140/2.1.3387.2969

[10] C. Walshaw, "A multilevel melodic similarity framework," in *Extended Abstracts for the Late-Breaking Demo Session of the 16th International Society for Music Information Retrieval Conference, 2015*, Málaga, Spain, 2015.

[11] A. Shahid, D. Diamond, and J. McDermott, "Patterns2KG: JAMS pipeline for modeling music patterns," in *International Joint Workshop on Semantic Web and Ontology Design for Cultural Heritage*, 2023. [Online]. Available: https://api.semanticscholar.org/CorpusID:266377142

[12] J. de Berardinis, V. A. Carriero, N. Jain, N. Lazzari, A. Meroño-Peñuela, A. Poltronieri, and V. Presutti, "The Polifonia Ontology Network: Building a semantic backbone for musical heritage," in *International Semantic Web Conference*. Springer, 2023, pp. 302–322.

[13] V. Carriero, J. de Berardinis, A. Meroño-Peñuela, A. Poltronieri, and V. Presutti, "The Music Meta Ontology: A flexible semantic model for the interoperability of music metadata," in *ISMIR 2023 Hybrid Conference*, 2023.

[14] J. De Berardinis, A. Meroño Peñuela, A. Poltronieri, and Presutti, "The Music Annotation Pattern," in *WOP2022: 13th Workshop on Ontology Design and Patterns*, Hangzhou, China, 2022.

[15] P. Tovstogan, X. Serra, and D. Bogdanov, "Visualization of deep audio embeddings for music exploration and rediscovery," in *Proceedings of the 19th Sound and Music Computing Conference*, 2022, pp. 493–500.

[16] P. Knees, M. Schedl, T. Pohle, and G. Widmer, "An innovative three-dimensional user interface for exploring music collections enriched," in *Proceedings of the 14th ACM International Conference on Multimedia*, ser. MM '06. New York, NY, USA: Association for Computing Machinery, 2006, p. 17–24. [Online]. Available: https://doi.org/10.1145/1180639.1180652

[17] S. Abdallah, E. Benetos, N. Gold, S. Hargreaves, T. Weyde, and D. Wolff, "The digital music lab: A big data infrastructure for digital musicology," *J. Comput. Cult. Herit.*, vol. 10, no. 1, jan 2017. [Online]. Available: https://doi.org/10.1145/2983918

[18] C. Walshaw, "A visual exploration of melodic relationships within traditional music collections," in *2018 22nd International Conference Information Visualisation (IV)*, 2018, pp. 478–483.

[19] ——, "Constructing proximity graphs to explore similarities in large-scale melodic datasets." in *6th Intl Workshop on Folk Music Analysis*, 2016, pp. 22–29.

[20] ——, "TuneGraph, an online visual tool for exploring melodic similarity," in *Proc. Digital Research in the Humanities and Arts*, 2015. [Online]. Available: https://api.semanticscholar.org/CorpusID:2605457

[21] B. Duggan and B. O'Shea, "Tunepal - disseminating a music information retrieval system to the traditional Irish music community," in *Proceedings of the 11th International Society for Music Information Retrieval Conference, ISMIR 2010, Utrecht, Netherlands, August 9-13, 2010*, J. S. Downie and R. C. Veltkamp, Eds. International Society for Music Information Retrieval, 2010, pp. 583–588. [Online]. Available: http://ismir2010.ismir.net/proceedings/ismir2010-100.pdf

[22] B. Duggan, "Tunepal: the traditional musician's toolbox," in *Proceedings of the Second Workshop on EHeritage and Digital Art Preservation*, ser. eHeritage '10. New York, NY, USA: Association for Computing Machinery, 2010, p. 25–30. [Online]. Available: https://doi.org/10.1145/1877922.1877931

[23] B. Duggan and B. O'Shea, "Tunepal: Searching a digital library of traditional music scores." *OCLC Systems & Services*, vol. 27, 10 2011.

[24] J. McDermott, D. Diamond, A. Shahid, A. Poltronieri, A. Meroño Peñuela, R. Fournier-S'niehotta, and T. Zhu, "Polifonia deliverable d3.2: Analysis of music repositories to identify musical patterns," Polifonia Consortium, Tech. Rep., 2021, https://polifonia-project.eu/dissemination/.

[25] A. Volk and P. Van Kranenburg, "Melodic similarity among folk songs: An annotation study on similarity-based categorization in music," *Musicae Scientiae*, vol. 16, 2012.

[26] B. Janssen, P. van Kranenburg, and A. Volk, "Finding Occurrences of Melodic Segments in Folk Songs Employing Symbolic Similarity Measures," *Journal of New Music Research*, vol. 46, no. 2, pp. 118–134, Apr. 2017. [Online]. Available: https://www.tandfonline.com/doi/full/10.1080/09298215.2017.1316292

[27] J. de Berardinis, A. Meroño-Peñuela, A. Poltronieri, and V. Presutti, "ChoCo: a chord corpus and a data transformation workflow for musical harmony knowledge graphs," *Scientific Data*, vol. 10, 09 2023.

[28] J. de Berardinis, A. Meroño Peñuela, A. Poltronieri, and V. Presutti, "The Harmonic Memory: a knowledge graph of harmonic patterns as a trustworthy framework for computational creativity," in *Proceedings of the ACM Web Conference 2023*, ser. WWW '23. New York, NY, USA: Association for Computing Machinery, 2023, p. 3873–3882. [Online]. Available: https://doi.org/10.1145/3543507.3587428

[29] N. Lazzari, A. Poltronieri, and V. Presutti, "Pitchclass2vec: Symbolic music structure segmentation with chord embeddings," in *Proceedings of the 1st Workshop on Artificial Intelligence and Creativity co-located with 21th International Conference of the Italian Association for Artificial Intelligence (AIxIA 2022)*, 2022, pp. 14–30. [Online]. Available: https://api.semanticscholar.org/CorpusID:254223385