Capturing the Meaning of Complex Texts about Music

Sutcliffe, Richard; Liem, Cynthia

Publication date
2017

Document Version
Final published version

Citation (APA)
Sutcliffe, R., & Liem, C. (2017). *Capturing the Meaning of Complex Texts about Music*. 1-2. Abstract from ISMIR 2017, Suzhou, China. https://ismir2017.smcnus.org/lbds/Sutcliffe2017.pdf

Capturing the Meaning of Complex Texts about Music

Richard Sutcliffe
School of CSEE
University of Essex, UK
rsutcl@essex.ac.uk

Cynthia Liem
Multimedia Computing Group
TU Delft, The Netherlands
c.c.s.liem@tudelft.nl

ABSTRACT

Following four annual C@merata evaluations at MediaEval, in which natural language phrases about music must be mapped to passages in MusicXML scores, we have developed a representation which captures the meaning of a text about music as a JSON feature structure. We have written a system to convert any text to this representation, comprising 23 stages of pre-processing, followed by application of the SpaCy statistical dependency parser. We have applied the system to the C@merata test queries with accurate results. Our approach can capture extremely detailed information, as well as general musical ideas.

1. INTRODUCTION

We are interested in linking detailed texts about music to the actual passages they refer to. For example, 'C sharp in the treble clef' refers specifically to a note on a particular clef while 'Thor's Hammer' is the informal name for a theme in the last movement of Sibelius' 5th Symphony. Such references occur frequently in musicological books, internet pages and so on [5]. Text was one of Downie's Seven Facets of Music [1] but it has not been studied in detail before. Most previous NLP work on music has been concerned with song lyrics, starting with [2], while Oramas et al. concentrate on conventional Named Entities and relations between them [3] (see [7] for a detailed review of all work combining NLP and music).

C@merata[1] is an annual evaluation which commenced in 2014 [4,6]. Each year there are 200 questions comprising a noun phrase (e.g. 'dotted crotchet Bb in the right hand in bars 23-40') and a MusicXML[2] score (e.g. Scarlatti K30). In the example case, a note of the specified length and pitch must be found, played by the specified hand and falling within a range of bars.

Attempting the C@merata task involves extracting all information from the query (by NLP) and searching for matching passages in a score (by MIR). This paper is concerned with the first step and more details can be found in the poster [8].

[1] http://csee.essex.ac.uk/camerata/
[2] http://www.musicxml.com

2. CONVERTING TEXT TO JSON

We have developed an initial system whose input is a noun phrase and whose output is a JSON feature structure capturing all the musical information in the phrase. The system has been developed from the 2014-2016 C@merata data sets [4,6,7] which contain very complex phrases, some taken from musicological texts, some created according to a defined distribution of question types. It is based on the SpaCy parser[3] and starts with tokenisation followed by Part-of-Speech tagging. We then carry out a series of pre-processing steps, before parsing takes place. Finally, the parse tree is traversed to extract the final JSON corresponding to the input.

SpaCy is a statistical parser which returns a dependency-based parse tree for any input sentence. It is highly accurate on general texts, but in the musical domain it needs to be augmented with specific terminological and phraseological information; otherwise the extremely complicated noun compounds found in musical texts are not analysed correctly and attachments between constituents (e.g. linking a prepositional phrase to the head noun phrase) are therefore not correct either.

Moreover, it is necessary to make explicit the exact meaning of musical terms; for example 3/4 is likely to be a time signature not a general fraction, 'Ab' is an A flat and is not a proper name, despite starting with a capital letter, and 'passage' is likely to refer to a portion of music not to the voyage of a ship.

```
G# quaver in the right hand against a crotchet
in the left hand in bars 1-25
{
  "first": {                    "second": {
    "note_accidental": 1,         "measure_from": 1,
    "note_divisions": 48,         "measure_to": 25,
    "note_length": 24,            "note_divisions": 48,
    "note_name": "g",             "note_length": 48,
    "note_octave": -1,            "staff_hand": "left"
    "staff_hand": "right"       },
  },
  "type": "against"
}
```

Figure 1. Conversion of a simple query into JSON

Each of our steps handles one type of musical term or phrase. At each stage, we use a top-down recursive-descent parser to recognise instances of a particular phrase type in the input and to replace each with a non-terminal symbol. These non-terminals can then form part

[3] https://spacy.io/

of the analysis of higher-level music terms in later stages. When we reach the final SpaCy parsing, the important phrases are already recognised and in consequence cannot be accidentally split by the parser. Essentially, we uncover the structure of musical terminology, going down to the smallest possible details which a text can refer to.

To understand our approach, consider Figure 1 which shows a sample query and its output. 'G# quaver in the right hand' becomes a set of attribute-value pairs which capture the length of the note (24/48 of a crotchet, i.e. a quaver or eighth note), its name ('g'), its octave pitch ('-1' i.e. unknown) and the hand playing it ('right'). Similarly, 'a crotchet in the left hand in bars 1-25' becomes a range of bars ('1' to '25') a length (48/48 i.e. a crotchet or quarter note), and a hand ('left'). Finally, we see that the two are playing together ('against' in the query) so the query has type 'against' and is in two parts 'first' and 'second'. Both European ('crotchet', 'bar') and American ('quarter note', 'measure') musical terminology can be handled interchangeably.

3. PROCESSING STAGES

As we have stated, a top-down parser makes 23 passes over the input, looking for specific classes of musical construct. These are: (1) textures ('two-part texture'), (2) keys ('C major'), (3) notes ('slurred double whole note trill', 'D4' or just 'note'), (4) note sequences ('C#4 D4' or just 'melody'), (5) measures ('bars 1-10'), (6) word underlay ('on the word "Der"', (7) quoted instrument types ('"cello"'), (8) the music staff if specified ('treble clef', 'left hand'), (9) instruments (unquoted) ('violin I divisi' - we can specify which group of instruments 'I' and also performance indications relating specifically to the instruments 'divisi'), (10) phrases specifying several instruments ('cellos and double basses'), (11) triads in different inversions ('Ib triad'), and (12) intervals whether harmonic or melodic, perfect, imperfect, augmented or diminished ('doubly diminished harmonic fifth').

We then have (13) interval sequences ('alternating fourths and fifths'), (14) cadences ('interrupted cadence', 'perfect cadence'), (15) Prepositional phrase chord inversion ('in the first inversion'), (16) chords ('chord of F#3, D4 and A4', 'D minor chord', or just 'chord'), (17) arpeggios ('F sharp minor arpeggio', 'ascending arpeggio'), (18) scales ('C major scale', 'minor scale' or just 'scale'), (19) melodies ('five-note melody', 'melody'), (20) event sequences ('descending arpeggio in quavers *followed by* ascending arpeggio in quavers'), (21) synchronous events ('rocking eighth-note chords in the piano right hand *against* half-note octaves in the piano left hand'), (22) detached performance indications ('*fermata* on a whole note'), and (23) time signature ('12/8', '2').

SpaCy parsing takes place next; we then join prepositional phrases to the head noun phrase: 'crotchet C4 in the bass clef on the double basses' is one JSON specifying a note in a particular clef and played by a stated instrument.

We not only extract details; we can also capture general ideas and partial specifications as well ('passage', 'note', 'arpeggio', 'chord' etc.). Concerning metaphorical usage (e.g. Deryck Cooke on Bruckner: 'hammering ostinatos' or Antony Hopkins on Beethoven: 'decked with garlands of scales from flutes, clarinets and bassoons' we can extract the specific information and store the metaphors for later processing; here we see that 'hammering' is an unknown property of an ostinato and that 'decked' and 'garlands' are connected with the scales in some way.

4. CONCLUSION AND NEXT STEPS

This initial experiment was quite successful; highly complex and detailed information can be captured accurately, alongside general and metaphorical language. Our next steps are to extend the work to a larger range of texts, to carry out a thorough evaluation, and to incorporate the module into an MIR system.

5. REFERENCES

[1] J. S. Downie: Music Information Retrieval. *Annual Review of Information Science and Technology*, 37:295-340, 2003.

[2] B. Logan, A. Kositsky, and P. Moreno: "Semantic analysis of song lyrics," *IEEE ICME 2004*.

[3] S. Oramas, L. Espinosa-Anke, M. Sordo, H. Saggion, and X. Serra: "Information Extraction for Knowledge Base Construction in the Music Domain," *Journal of Knowledge & Data Engineering*, 106, 70-83, 2016.

[4] R. Sutcliffe, T. Crawford, C. Fox, D. L. Root, and E. Hovy: "Shared Evaluation of Natural Language Queries against Classical Music Scores: A Full Description of the C@merata 2014 Task," http://csee.essex.ac.uk/camerata/newest_camerata_long_14.pdf.

[5] R. Sutcliffe, T. Crawford, C. Fox, D. L. Root, E. Hovy, and R. Lewis: "Relating Natural Language Text to Musical Passages," *Proc. 16th ISMIR 2015* http://ismir2015.uma.es/articles/ 263_Paper.pdf.

[6] R. Sutcliffe, D. O Maidin, and E. Hovy: "The C@merata task at MediaEval 2017: Natural Language Queries about Music, their JSON Representations, and Matching Passages in MusicXML Scores," Proc. MediaEval 2017, Dublin.

[7] R. Sutcliffe, E. Hovy, T. Collins, S. Wan, T. Crawford, and D. L. Root: "Searching for Musical Features using Natural Language Queries: The C@merata Evaluations at MediaEval," *Language Resources and Evaluation Journal* (to appear).

[8] R. Sutcliffe and C. Liem: Poster. http://csee.essex.ac.uk/staff/rsutcl/ismir2017.pptx.