TEXT INFORMATION RETRIEVAL APPROACH TO MUSIC INFORMATION RETRIEVAL

Jacek Wolkowicz
Faculty of Computer Science
Dalhousie University
jacek@cs.dal.ca

Vlado Kešelj
Faculty of Computer Science
Dalhousie University
vlado@cs.dal.ca

ABSTRACT

This MIREX submission for symbolic music similarity task introduces textual information retrieval thinking into the process of music information retrieval. The main contribution of this approach is to utilize well established term weighting methods for text retrieval and check their suitability for music data, which is a second reason for this submission apart from the obvious one - to compete with other algorithms in symbolic music similarity task. We use a simple feature extraction method, so that the performance of an algorithm depends only on the applied term weighting function. The parameters for each of the algorithms are optimized based on 2005 SMS MIREX data.

1. INTRODUCTION

It has been show in the previous releases of MIREX SMS (Music Information Retrieval Evaluation eXchange, Symbolic Music Similarity) task, that bag-of-words methods, which are well established approaches in textual Information Retrieval, work well for retrieving similar melodies from music corpora. What is noticeable, is that the main focus is paid to how to transfer the input sequence of notes into a set of features and how to compare (measure the similarity) between different feature sets.

However, in textual information retrieval it has been found that a simple word extraction (as feature extraction) and basic similarity measures (like cosine similarity) is enough. The really important thing though is term weighting, i.e. given documents documents in a dataset - to determine which terms are more important for each document or the dataset as a whole. Some frequent terms (dubbed stopwords) don't usually even take part in the retrieval process at all. This reduces retrieval time, but primarily - allows for better ordering of the retrieval results, which is also the main point of MIREX SMS task.

2. BACKGROUND

The approach to symbolic music retrieval proposed in this paper focuses on evaluation of different term weighting methods. Other parts of the retrieval process we kept rather standard.

The process of document retrieval starts with indexing features extracted from the corpus documents. Input dataset consists a set of standard MIDI files, each containing a single track of notes representing a monophonic melody. Since none of the notes are concurrent or overlap, string based methods can be easily applied to the input documents. Like text documents that can be just seen as series of characters, monophonic music opi are just series of notes. The difference with music files is that text documents are easily separable into basic features - words. Since there is no such a thing as a clear phrase boundary in music, the usual bag-of-words, or bag-of-terms approach consists of building n-grams, i.e. substrings of n consecutive tokens (notes) that start with every note. This process is widely used also in bio-informatics (DNA sequence analysis) and in some text processing tasks as well (for authorship attribution [1] or for tasks with languages with no word boundaries, like thai).

Each of the features that conforms an n-gram, which we call it here, a uni-gram, is derived from each note event that one finds in the notes stream. It can either contain absolute values representing music features, such as note's pitch, duration or IOI, but in most cases relative (interval) features are used. We went with the last approach using either melodic intervals or a combinations of melodic and IOI intervals. These features (or combinations of features) gave us the best performance for various other settings when we have tested them on the released 2005 MIREX SMS dataset.

With this test we were also able to determine the optimal n for each of the proposed algorithms. It varies from 2 to 5, depending on the parameters. The general rule of thumb is though, the more general the features, the bigger the n

should be.

We have also tested various similarity measures figuring out that a simple cosine similarity, widely used in textual information retrieval, gives good results with music data. The basic formula for the cosine similarity is the following:

$$sim(\mathbf{x}, \mathbf{y}) = \frac{\sum_i x_i y_i}{\sqrt{\sum_i x_i}\sqrt{\sum_i x_i}} \tag{1}$$

where x_i is a weight of a term i in a document $\mathbf{x}$. The main contribution of this submission is how different, text-based term weighting measures affect music information retrieval.

3. ALGORITHMS

3.1 Binary weights (WK1)

The WK1 algorithm uses a basic binary term weighting approach. It is either 0 (if a term, or an n-gram doesn't appear in the document) or 1 (if a term appears in the document), so the resulting similarity measure between two documents is the number of terms in common normalized by geometric average of numbers of unique terms in both documents. This algorithm got the best results on 2005 SMS MIREX dataset for melodic intervals used as features and n being 5.

3.2 Term count weights (WK2 and WK3)

The following two algorithms use simple term counts (the number of times the term appears in the compared documents) which gives a classical cosine similarity definition. This rather simple method gave us surprisingly good results for two different settings so we have decided to submit both for the competition. The first one (WK2) uses again melodic intervals as simple features and n 4, while the second one uses a combinations of melodic and IOI intervals with n 2. The relative performance of these two algorithms will allow to assess whether introduction of rhythmic features helps to improve the overall score.

3.3 tf.idf term weights (WK4)

WK4 algorithm computes standard tf.idf weights of each term from documents to compare, which gives each term i a weight depending on its count (c_i) within the document (d) and in how many documents of the collection (D) a term i occurs. The formula is given as follows:

$$tf.idf_i = \frac{c_i}{\|d\|} \log \frac{\|D\|}{\delta_i} \tag{2}$$

where $\delta_i = \|\{d \in D | i \in d\}\|$ is the number of documents containing term i. This measure is commonly used in Textual Information Retrieval for term weighting so it would be interesting, how it performs in music challenge. For the settings of WK4 we have again determined, that n equals 4 and melodic interval features worked the best for 2005 SMS data.

3.4 Okapi BM25 (WK5, WK6)

BM25, unlike tf.idf, is an industry-developed weighting scheme, that outperforms classic term weighting measures, like tf.idf. It tries to capture roughly the same concept as original tf.idf measure but tries to balance documents with different lengths and different term distribution:

$$bm25_i = \frac{c_i(k+1)}{c_i + k(1 - b + b\frac{\|D\|}{avgdl})} \log \frac{\|D\| - \delta_i + 0.5}{\|D\| + \delta_i} \tag{3}$$

where $avgdl$ is an average document length. It is parametrized, with parameters b and k, and we have used a recommended setting of $b = 0.75$ and $k = 2$. Since it should be a top performing function, we have came up with two sets of settings: WK5 with melodic interval features of length 4 and WK6 with features combining melodic interval and IOI ratios with n-gram length of 2.

4. REFERENCES

[1] Vlado Keselj, Fuchun Peng, Nick Cercone, and Calvin Thomas. N-gram-based author profiles for authorship attribution. In *Proc. of the PACLING03 Conf.*, pages 255–264, 2003.