Music Composer Identification

MUS-15

Maureen Tanuadji

29.06.2015

Abstract

This report shows different possibilities to identify the composer of a music piece and how N-gram model based on Natural Language Processing and Information Retrieval can be used to solve the identification problem.

Introduction

The studies to identify the composer of a music piece has been developed for years under the name of stylometry studies. An expert in stylometry analyzes the music piece and uses "fingerprints" of the composer which is called as the style marker to do the classification [1]. Since the style can be seen as a recurring arrangement of features, it is very intuitive that we try to solve the composer identification problem with a recognition algorithm [3].

Furthermore, it has also been discovered that the starting point of creating music is concurrent with language development. It also implies that music can be seen as a natural language. This idea inspires Wolkowicz, Kulka and Kešelj [7] to apply the already developed methods being used to solve the authorship attribution problem from Natural Language Processing and Information Retrieval into the composer recognition task.

Previous Work

Based on the explanation on Wolkowicz, Kulka and Kešelj [7], several systems have been developed in the composer recognition area. Pollastri and Simoncelli [6] have developed a system of theme recognition using Hidden Markov Model. This research is done only with monophonic themes and they report 42% accuracy among 5 composers. Buzzanca [2] has developed a successful style recognition system using supervised neural network, where he reports a 97% accuracy. However, this solution is not considered fully automated because it involves expert work on data preprocessing. In addition, there is also a drawback from the use of neural networks. The relation between the behavior of the recognition process and recognition result can not be easily observed.

A system that successfully solves the authorship attribution problem on texts has been developed by Kešelj, Peng, Cercone, and Thomas [5]. They show that N-gram-based statistical approach from Natural Language Processing can be applied to texts and it can reach 100% accuracy. The implemented method is simple and might be successfully applied to other fields, such as music. In the N-gram-based approach, it is important to assume that the order of notes plays a role to distinguish the composer. For an unknown composer's music piece, a profile will be extracted and checked based on the similarity of the existing features among all known profiles in order to determine a certain composer's attribution.

N-gram-based Approach

In this section the working process of the N-gram-based approach will be explained. First, the MIDI files will be preprocessed and N-grams information will be extracted to build the composer profile. The process continues with comparing the profile of unknown and known composer and calculating the similarity score to identify the composer.

Musical data representation

The data representation is an important aspect of the algorithm. It determines the type of informations or features that can be processed by the system. There are three main different data sources which influence the approaches for composer identification [4]:

1. Raw audio – recorded sound stored as audio formats e.g. WAV. Possible extracted features from this data are cepstral coefficients, pitch class profiles, etc.

2. Symbolic representation – score notation stored in a certain application format e.g. mus in Finale or using MIDI protocol. Pitch and duration of the notes, harmony, and dynamics can be extracted as features from the file.

3. Metadata – stored meta information of the data, for example year/period, title, genre, instrumentation, etc.

The melodic and rhythmic information are quite easily obtainable from music data based on the symbolic music representation. They are also free from sound recording noises, therefore, the N-gram-based approach focuses to this representation especially on the MIDI files. They may also behave like textual files, they are easy to store, edit, and process.

The MIDI files that are used in the process are freely available on the internet. They consist of 5 different composers and only piano works are chosen for better compatibility. The composer, number of pieces, and file size are listed in the Table 1 below.

Table 1: MIDI corpus properties

	Composer	Training Set	Testing Set
1	J.S.Bach	99 items, 890 kB	10 items, 73 kB
2	L.van Beethoven	34 items, 1029 kB	10 items, 370 kB
3	F.Chopin	8 items, 870 kB	10 items, 182 kB
4	W.A.Mozart	15 items, 357 kB	2 items, 91 kB
5	F.Schubert	18 items, 863 kB	5 items, 253 kB

MIDI files also consist of several channels and tracks, which may overlap each other with respect to a timeline and several notes may occur at the same time on each channel. For that reason, each channel will be treated separately and each channel only correspond to one staff (or hand). Further constraint is also needed to solve the overlap problem. In each channel, only the highest currently played note is taken.

N-gram Profile Extraction

The features of N-gram profile consist of:

1. *Melodic* – Each melodic (pitch) is stored in MIDI in the form of MIDI key number

2. *Rhythmic* – Rythmic give the information about the duration for each note in beats per minute (BPM)

3. *Melodic and rhythmic combined* – The combination of melodic and rhythmic can be used to build one additional unique feature

The first step of the N-gram extraction is to find the unigram representation from the prepared MIDI data. However several adjustments have to be made to get good feature representations. The pitch has to be key independent because key does not give any influence in the identification process therefore we only consider the relative pitches, i.e. the difference of our current pitch to the next pitch. A relative duration counting in a logarithmic scale has also been used as the feature instead of the real duration value. The formula to extract the features is given as follows:

$$(P_i, T_i) = \left(p_{i+1} - p_i, round \left(\log_2 \left(\frac{t_{i+1}}{t_i} \right) \right) \right), \quad (1)$$

where p_i denotes the i-th note pitch, t_i denotes the i-th note duration and (P_i, T_i) is the resulting tuple.

The process for transforming unigrams into N-grams is straightforward. It is done by concatenating N consecutive unigrams into one item. As the result, three different features can be retrieved based on the N-grams, namely the N-gram for melody only, rhythm only and the combination of both representation. An example process of the profile extraction with N=3 can be seen in Fig. 1

Composer Recognition Task

From the profile which is formed by N-grams, comparisons with the appropriate profiles of other composers are done using the following similarity measure (it is a modified method described by Kešelj, Peng, Cercone, and Thomas [5]):

$$Sim(x, y) = \sum_i \left(4 - \left(\frac{2 \cdot (x_i - y_i)}{x_i + y_i} \right)^2 \right), \quad (2)$$

where x and y stands for a composer profile and the corresponding profile of a piece. For each composer, 3 similarity values will be calculated. In order to find the most probable composer for the piece, the following steps are applied:

1. Sum up all the similarities for each composer profile.

2. Sort all sums descending.

3. Take a composer with the highest sum as the result.

The example evaluation calculations are shown in Table 2 and Table 3.

Table 2: Evaluation of the Frederic Chopin prelude Op. 28 No. 22

		Profiles			Total	Verdict
		melodic	rhythmic	combined		
Composer	Beethoven	43.2	17.2	11.0	71	3
	Mozart	49.2	11.4	6.4	67	4
	Bach	62.4	8.2	6.4	77	2
	Schubert	19,3	13.2	5.9	38	5
	Chopin	86.8	25.1	10.9	122	1

Table 3: Evaluation of the Ludwig van Beethoven Sonata Op. 49 No. 2

		Profiles			Total	Verdict
		melodic	rhythmic	combined		
Composer	Beethoven	303.7	208.8	109.7	622	4
	Mozart	319.2	201.7	124.4	645	2
	Bach	366.1	263.0	83.7	712	1
	Schubert	315.6	201.8	119.1	636	3
	Chopin	296.5	127.3	79.0	502	5

Table 2 shows a proper judgement from the system for the music pieces of Frederic Chopin, Prelude Op. 28 No. 22. It is correctly identify the same composer because the unknown composer's style resemblances Frederic Chopin style as we can see from high similarity score result. However, the algorithm might produces poor judgment if the composition style of the composer differs from the composer's common style. The example of this condition can be seen in Table 3. The algorithm identifies the Beethoven Sonata Op. 49 No. 2 belongs to Bach instead of Beethoven.

Results

There are several degrees of freedom in the system, e.g. N-gram length (N) and profile size. An aging factor as the representation of composer's era can also be introduced to enhance the identification process. Table 4 shows the various combination of N-gram lengths and profile sizes using a constant aging factor (0.96). The accuracy of the system (i.e. the system correctly assigning pieces to the composer in the test collection) reaches 84% for the combination of profile

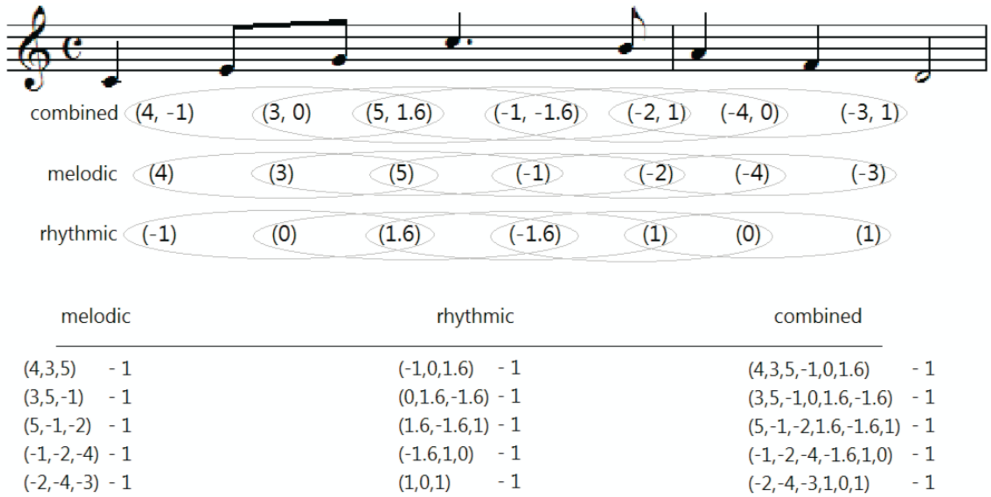

Figure 1: Building profiles from a tune

Table 4: Results of the algorithm.

N \ size	100	250	500	1000	2500	5000	10000
2	0.41	0.38	0.38	0.35	0.32	0.43	0.43
3	0.46	0.54	0.59	0.62	0.59	0.51	0.43
4	0.62	0.70	0.65	0.73	0.73	0.78	0.86
5	0.54	0.62	0.70	0.78	0.78	0.81	0.81
6	0.54	0.59	0.68	0.68	0.84	0.78	0.84
7	0.46	0.49	0.68	0.68	0.68	0.70	0.84
9	0.46	0.57	0.49	0.51	0.57	0.68	0.76
12	0.41	0.46	0.41	0.41	0.41	0.46	0.49

size=2500 and N=6. It also reaches 86% with the combination of profile size=10000 and N=4.

Accuracies over 80% for both combinations, longer N-gram length with smaller profile size or smaller N-gram length with bigger profile size, can be seen as good results, while the random classifier can obtain only 20% accuracy. The second important remark is the fact that some pieces were written by the composer in a different style, therefore, they are quite difficult to be classified in the proper class even by common people. That is why the algorithm might not reach 100% accuracy.

Conclusions

The analysis shows that Natural Language Processing and Information Retrieval tools can be use to analyze music. It also shows that the other methods, e.g. clustering, plagiarism detection, and much more can be further developed for music processing. The N-gram-based approach is proven to be useful in solving the problem of composer identification.

References

[1] E. Backer and P. van Kranenburg. On musical stylometry—a pattern recognition approach. *Pattern Recognition Letters*, 26(3):299–309, 205.

[2] G. Buzzanca. A supervised learning approach to musical style recognition. In *Music and Artificial Intelligence. Additional Proceedings of the Second International Conference, ICMAI*, volume 2002, page 167. Citeseer, 2002.

[3] M. Hontanilla, C. Pérez-Sancho, and J. Iñesta. Modeling musical style with language models for composer recognition. In J. Sanches, L. Micó, and J. Cardoso, editors, *Pattern Recognition and Image*

Analysis, volume 7887 of *Lecture Notes in Computer Science*, pages 740–748. Springer Berlin Heidelberg, 2013.

[4] M. Hontanilla, C. Pérez-Sancho, and J. M. Inesta. Composer recognition using language models. *Proc. of Signal Processing, Pattern Recognition, and Applications (SPPRA 2011)*, pages 76–83, 2011.

[5] V. Kešelj, F. Peng, N. Cercone, and C. Thomas. N-gram-based author profiles for authorship attribution. In *Proceedings of the conference pacific association for computational linguistics, PACLING*, volume 3, pages 255–264, 2003.

[6] E. Pollastri and G. Simoncelli. Classification of melodies by composer with hidden markov models. In *Web Delivering of Music, 2001. Proceedings. First International Conference on*, pages 88–95. IEEE, 2001.

[7] J. Wołkowicz, Z. Kulka, and V. Kešelj. N-gram-based approach to composer recognition. *Archives of Acoustics*, 33(1):43–55, 2008.