World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:3, No:7, 2009

Exploring Performance-Based Music Attributes for Stylometric Analysis

Abdellghani Bellaachia, Edward Jimenez

***Abstract*—**Music Information Retrieval (MIR) and modern data mining techniques are applied to identify style markers in midi music for stylometric analysis and author attribution. Over 100 attributes are extracted from a library of 2830 songs then mined using supervised learning data mining techniques. Two attributes are identified that provide high informational gain. These attributes are then used as style markers to predict authorship. Using these style markers the authors are able to correctly distinguish songs written by the Beatles from those that were not with a precision and accuracy of over 98 per cent. The identification of these style markers as well as the architecture for this research provides a foundation for future research in musical stylometry.

***Keywords*—**Music Information Retrieval, Music Data Mining, Stylometry.

I. Introduction

FOR centuries researchers have analyzed the style of authors to help authenticate or assign attribution to written works. This type of analysis, known as Stylometry, relies on the fundamental principle that the essence of an author's individual style can be captured and quantified. In the late 19th century, this form of analysis became increasingly quantitative and mathematical. Thomas Mendenhall, a self-taught physicist, compared the frequency distribution of words of differing length in the writings of authors William Shakespeare and Lord Chancellor Francis Bacon, to argue that Bacon could not have written plays in Shakespeare's name as some had thought. Similar types of analysis have been used for author attribution of the Federalist Papers, and more recently, to (successfully) predict the true identity of the anonymous author of the novel, Primary Colors [1],[8].

Stylometric analysis has historically taken on more qualitative than quantitative measures. In this experiment however, the authors test their hypothesis that with the tools available today, quantitative measures can be a focus of stylometric analysis. This follows the premise that it is most important to discover that measurable differences in style exist; there may not be interesting meaning behind those differences apart from the fact that they are statistically significant. Furthermore, the early attribution of meaning or significance to style markers may introduce bias that hinders later evaluation.

While the primary corpus in the field of Stylometry describes the evaluation of written text, over the last decade or so examples of stylometric analysis in the musical domain have become abundant. Just as early students of stylometry, often students of literature, history, etc. used their domain knowledge to understand style, early attempts of music stylometry also relied upon an understanding of the musical domain. One method of early music stylometry used variations in songs' written form to identity authorship. For example, two composers may prefer different ways of writing the same musical passage. This type of stylistic difference was used to tell one composer's musical score from another [2],[6],[7].

More recently, a well-known Web-based service, Pandora, relies on stylometry. After users list their favorite songs, the service plays songs that are similar in style. This is accomplished using work from the Human Genome Project, for which musicians analyzed and scored songs from over 150 genres and for hundreds of attributes such as "Empowering Lyrics" or "Aggressive Drumming" [3].

As members of the computing and data mining community rather than musicians and music theorists, the authors take a more quantitative approach. Rather than attempting to identity style markers intuitively, as a music theorist might, an empirical determination of those markers is made using data mining techniques. The markers are then used to determine whether or not a song was written by the Beatles.

II. Methods and Materials

A. Research Approach

The initial step in this experiment is to build a large collection of music. The music data (songs) is then cleansed and manually separated into two classes: those songs written by the Beatles (B) and those that are not (B'). Using music feature extraction tools, a dataset (S) is created using candidate style markers. The resulting dataset is defined as:

$$S = \sum_{x}^{1} f_x(B) + \sum_{x}^{1} f_x(B') \quad \text{x=101} \qquad (1)$$

The style features are loaded into the test database then used to build a classification model that is capable of determining the class (B or B') of the midi file.

A. Bellaachia is with the George Washington University Department of Computer Science, George Washington University, Washington DC 20052, USA. (phone: 202-994-8166; e-mail: bell@gwu.edu).

E. Jimenez was a graduate student at George Washington University Department of Computer Science. He is now a Senior Software Engineer at Trusted Concepts Inc., Chantilly, VA 20151, USA. (e-mail: ed.jimenez@trustedconcepts.com).

B. Architecture

The goal is to create a modular, extensible architecture that can be used with a range of existing tools rather than focusing on the results from a single tool. To do this, Pentaho Data Integration (PDI) platform is used. Transformations are created that load the output from the extraction tools into new attributes in a MySQL database. This architecture makes it easy to add and compare attributes across extraction tools.

Focusing on free, open-source software, in addition to PDI the tools selected include:

- Firefox and the Flashgot add-in to collect data manually from a variety of free midi collections on the internet,
- MuseScore for midi conversion to MusicXML,
- JSymbolic for feature extraction,
- Humdrum for feature extraction,
- Uwin, Cygwin, Ubuntu Unix platforms for extraction tool testbed,
- Weka data mining and visualization software for feature selection.

After a series of tests, the authors were unable to successfully convert the entire midi dataset to the Humdrum format. Therefore, that tool was removed from the test bed and further experiments rely wholly on JSymbolic for feature extraction.

C. Data

Music data in midi format [4] is used for this experiment. Midi files contain instructions for creating the sound rather than an electronic representation of the sound itself. A detailed explanation of the format is beyond the scope of this paper. However, it is worth noting that the format was selected for several reasons:

- A wide-range of data is available. While there are several repositories for musical scores, most of these are for classical music. For this experiment it is critical to collect from a wide range of musical genres.
- Due to the simplified nature of midi files, the information is readily available for data mining. The midi format is widely accepted and can be used in most of the music analysis tools.
- Midi objectively describes the music – it does not leave fundamental and potentially important elements of the music such as pitch, intensity, and timbre open for interpretation.
- A focus is placed on performance representations of the music rather than written representations (scores).
- Midi file sizes are much more manageable. A full-length song in .mp3 format may be several Megabytes (Mb) where the same song represented in midi format is less then 100Kb. The cost of reduced sound quality is acceptable for this experiment.
- Midi music players are prevalent making it easy to put the stylometric determinations to the test with the human ear.

The test dataset consists of 2830 midi files across a spectrum of artists and genres. Of those, 2594 are not Beatles songs (B') and the remaining 236 songs were written by the Beatles (B).

Due to the lack of an available consolidated repository of midi music large enough for this experiment, The Internet was mined for midi music using the Firefox browser and the FlashGot add-in download manager. This allows the tedious aspects of file downloading to be automated while taking care to separate the files into their respective classes.

Cleansing the data is a two-step process. First, duplicate songs are manually removed. A second instance of a song is allowed to remain if it is discernibly different from the one already obtained. Then, the midi format of each is validated using MuseScore, JSymbolic, and control scripts.

Using JSymbolic, 101 features are extracted from each of the midi files collected. A list and description of all of the features available from JSymbolic are provided in McKay[5]. Multi-dimensional features are not used in this experiment.

For the transformation step, within PDI the XML data values produced by JSymbolic are loaded into the database and an attribute is added for each song defining the class (B or B') to which that song belongs.

D. Feature Selection

Because many classifiers in Weka automatically perform feature selection/reduction, the first attempts used the entire feature set. Some classifiers, most notably SVM, could not initially accept all attributes, so some classification runs were preprocessed through an attribute selection process.

After several passes of feature selection, using Weka's InfoGain attribute evaluator, subsets of the data were created that contained the best 2, 3, 5, 10, 15, 20 and all attributes. Each dataset is then tested against a wide range of classifiers and boosting methods. An attempt was made to compensate for bias in the data towards B' by using methods like the Synthetic Minority Oversampling TEchnique (SMOTE) to create additional B Class datapoints.

Fig. 1 shows the effect of the size of the attribute set on the sensitivity (solid line) and accuracy (dashed line) in early models.

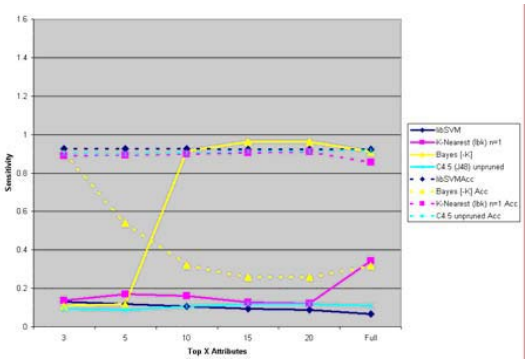

Fig. 1 Effect of Attribute Set Size on Sensitivity

III. Analysis and Results

Many classifiers were trained and tested in Weka using 10 fold cross validation. As expected tests showed an initial

inverse correlation between model sensitivity and accuracy. The plot in Fig. 2 shows a sample of the initial results for accuracy and sensitivity for some of the models tested including SVM, ADABoost, and K-Nearest Neighbor.

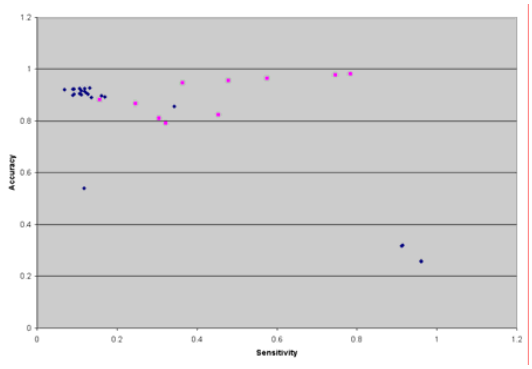

Fig. 2 Accuracy and Sensitivity

Table I shows the largely disappointing errors received in initial tests. The column provided gives the best performing result across all available datasets (specific size in parenthesis).

TABLE I
BEST-CASE PERFORMANCE SAMPLING OVER NON-COST CLASSIFIERS

	LibSVM (top 3)	K-Nearest (all)	C4.5 (top 15)	NaiveBayes (top 5)
Correctly Classified	92.7%	85.7%	91.4%	54.0%
Kappa Statistic	0.216	0.209	0.157	0.066
Mean absolute error	0.073	0.138	0.131	0.394
Root mean squared error	0.270	0.339	0.277	0.462
Relative absolute error	45.5%	89.9%	85.4%	257.1%
Root relative squared error	97.6%	122.7%	100.2%	166.9%

The data points in red (square) in Fig. 2 illustrate the success obtained once cost sensitive classifiers are used to compensate for the desire for greater scrutiny of the misses (false negatives). A cost of five is assigned for the false negatives resulting in the cost matrix shown in Table II.

TABLE II
COST MATRIX

0	5
1	0

Using this cost matrix while training new models has the most dramatic impact when the cost sensitive classifier is wrapped around the libSVM classifier. The best results are obtained using only two attributes:

- *MelodicOctaves* – The fraction of melodic intervals that are octaves, and
- *VoiceEqualityNumberofNotes* – The standard deviation of the total number of "Note On"s in each channel that contains at least one note.

The results from the most successful classifier model are shown in Table III. The resulting confusion matrix is shown below in Table IV.

TABLE III
COST-SENSITIVE LIBSVM CLASSIFIER RESULTS

Class	B	B′
TP Rate	.788	.999
FP Rate	.001	.212
Precision	.984	.981
Recall	.788	.999
F-Measure	.875	.990
ROC Area	.893	.893

TABLE IV
CONFUSION MATRIX FOR COST-SENSITIVE LIBSVM CLASSIFIER RESULTS

186	50
3	2591

IV. CONCLUSION AND FUTURE WORK

Using only two attributes, Beatles songs in the test dataset can be identified with an acceptable margin of error. While the results show promise for extracting and identifying style markers that support attribution of authorship, there is likely significant bias in the current model for this particular musical group (the Beatles) as well as their style of music (Western European Pop Music). The architecture used easily supports an expanded number of extraction tools, attributes and data. Follow on work will include an expanded test bed to evaluate style markers for a wider range of composers and musical genres.

REFERENCES

[1] H.Somers, "Stylometry and Authorship", School of Computer Science University of Manchester [Online]. Available: http://personalpages.manchester.ac.uk/staff/harold.somers/LELA30922/Authorship.ppt.

[2] A. Simoes, A. Louenco, and J. Joao Almeida, "Using Text Mining Techniques for Classical Music Scores Analysis", University of Minho, 2007.

[3] V. Vara, (2005, Oct. 6) "Unraveling Music's DNA," Wall Street Journal Online [Online]. Available: http://online.wsj.com/public/article/SB112784146741053451-FYfrFWfQ29np0rdqx33NDS8LtWM_20051105.html?mod=tff_article

[4] Midi Manufactures Association, "General MIDI 1, 2 and Lite Specifications" [Online]. Available: http://www.midi.org/techspecs/gm.php.

[5] C. McKay, (2004, June) "Automatic Genre Classification of MIDI Recordings", McGill University, Montreal.

[6] N Orio, "Music Retrieval: A tutorial and Review", Now Publishers Inc, 2006.

[7] Peter van Kranenburg, "Musical style recognition – a quantitative approach", University of Utrecht, Netherlands, Proceedings of the Conference on Interdisciplinary Musicology, 2004.

[8] E. Stamatatos, N. Fakotakis, and G. Kokkinakis, "Automatic Text Categorization in Terms of Genre and Author", University of Patras, Computational Linuistics, Dec. 2000.