META-MELO: A SYSTEM AND METHODOLOGY FOR THE COMPARISON OF MELODIC GENERATION MODELS

Nicolas Gonzalez Thomas, Philippe Pasquier, Arne Eigenfeldt, James B. Maxwell
MAMAS Lab, Simon Fraser University
ngonzale@sfu.ca, pasquier@sfu.ca

ABSTRACT

We investigate Musical Metacreation algorithms by applying Music Information Retrieval techniques for comparing the output of three off-line, corpus-based style imitation models. The first is *Variable Order Markov Chains*, a statistical model; second is the *Factor Oracle*, a pattern matcher; and third, *MusiCOG*, a novel graphical model based on perceptual and cognitive processes. Our focus is on discovering which musical biases are introduced by the models, that is, the characteristics of the output which are shaped directly by the formalism of the models and not by the corpus itself. We describe a system that implements the three models, along with a methodology for the quantitative analysis of model output, when trained on a corpus of melodies in symbolic form. Results show that the models output are indeed different and suggest that the cognitive approach is more successful at the tasks, although none of them encompass the full creative space of the corpus. We conclude that this methodology is promising for aiding in the informed application and development of generative models for music composition problems.

1. INTRODUCTION

Computational Musicology has generally focused on studying human composed music, however, algorithms for music generation provide a rich and relatively unexplored area for study. As algorithmic and generative models grow in number and complexity, the task of selecting and applying them to specific musical problems still remains an open question, for example, in the development of Computer Aided Composition (CAC) environments.

Stylistic Imitation, a particular Musical Metacreative approach [21], can be described as creativity arising from a pre-established conceptual space. This is at times referred to as Exploratory Creativity [5] and in practical musical terms is about generating new and original compositions that roughly cover the same space as the corpus, thus fitting a given musical style [2]. The conceptual space of a style can be defined by observing the musical features which remain invariant across the corpus.

The techniques applied for this task can be broadly categorized into two methodological groups: corpus based and non-corpus based methods. In the former, musical knowledge of the style is obtained through empirical induction from existing music compositions (generally in symbolic MIDI format), using machine learning techniques. Whereas in the latter this knowledge is provided by researchers in the form of theoretical and/or rule-based representations.

We are concerned here with applying Music Information Retrieval (MIR) tools in a controlled setting for the purpose of understanding more completely how these methods behave in real world applications. For this study we have chosen three corpus based models: the statistical *Variable Order Markov Model* (VOMM) [20], the *Factor Oracle* (FO) pattern matcher [8], and MusiCog [15], a novel, cognitively inspired approach used for the suggestion and contextual continuation (reflexive interaction) of musical ideas in the notation-based CAC system Manuscore [14].

The main question that we address is: given three corpus-based style-imitative models, which characteristics of the output are shaped by the underlying models themselves and not by the corpus? That is, we aim to discover the *musical biases* which arise from the formalism of the models. To answer this we investigate how each model's output is different in a *statistically significant* way.

A second question that arises is: what is the appropriate methodology for this research problem? We propose a framework and methodology for generating and evaluating melodies in a controlled setting where all models share the same fundamental conditions (Section 3.1). We use inter-model analysis to compare features from the melodic output of each model to the corpus and to the output of all other models, and intra-model analysis to reveal information about the relationships between the melodies generated by a single model.

(**add summary of results**)

Our contributions are: (1) a corpus of classical, popular, and jazz melodies, (2) *META-MELO* a MAX/MSP implementation of the three models, used for melodic generation from a corpus (Section 3), (3) a methodology which applies Machine Learning and MIR techniques for model output comparison (Section 5), and (4) the results of a study where we apply this methodology (Section 6).

Finally, we distinguish the tasks of composition from interpretation and are concerned here only with the former.

2. EXISTING WORK

We address the problem of music evaluation by using computational techniques to investigate model output in comparison to human-composed corpora, but also in terms of model self-consistency. This analysis is useful for determining which model output is truer to a corpus, and also for discovering more precisely how the models differ. Our hypothesis is that they differ to a degree that is statistically significant, and that this difference has an effect that is perceptible and can be described as a musical bias.

Manaris et. al. [13] use an artificial 'critic' based on power-law metrics as a fitness function for an evolutionary generative model, there is a computational evaluation of the similarity of generated melodies to the corpus. Begleiter et. al. [4] compare the output of different VOMM algorithms in the music domain, with the goal of measuring performance in terms of prediction; i.e., how closely a model imitates a particular style.

Very relevant to our work is the evaluation provided by Pearce and Wiggins [22] which empirically compares the output of three variants of Markov models trained on chorale melodies using musical judges. The stepwise regression also described provides directions for improving the models by indicating the quantifiable musical features which are most predictive in their failure.

Our approach empirically compares the output of three methodologically distinct corpus-based music generation models, without the intervention of human listeners. We also provide a simple technique for aiding the development of the models using decision trees.

(space
(space
space
space)

3. META-MELO

The system, implemented in MAX/MSP, is available for download together with the training corpus and a more detailed model description than is provided here [17]. We follow with a presentation of this component of the methodology presented in Figure 1.

3.1 Music Representation and MIDI Parsing

The system uses a simple but flexible representation for learning melodic (monophonic) music inspired by the multi-viewpoint approach proposed by Conklin and Witten [7]. The symbols used for the Markov model and Factor Oracle systems are derived from the concatenation of pitch interval and onset interval information. Several attributes can be used to train the models, which brings immediate challenges for the evaluation and comparison methodology. The approach we have chosen is to restrict the study and the description of the system, for the purpose of comparing the models in a controlled setting.

If the algorithms of the models are implemented in a simple form, we are likely to achieve a more transparent comparison, but with less interesting musical output, therefore reducing the value of the analysis. On the other hand, if more sophisticated implementations with musical heuristics are used for improving musical output, we will obtain results which are of poor generalization power with regards to the underlying models. It is worth noting that, in contrast to the corpus, none of the models contains an explicit model of tonality.

For the Markov and Factor Oracle implementations, the set of attributes is indexed and a dictionary is built for the set of symbols corresponding to that attribute. This way, when an event is observed which has not appeared before, a new entry is created in the dictionary. An important function in this initial stage is the *quantization* of temporal values to a reasonable minimum resolution of sixteenth notes. This allows the parsing function to: (1) group and classify similar events as the same element in the dictionary and (2) avoid creating entries with negligible differences. This, of course, would not be the case when dealing with human performance, where rhythmic variation based on interpretation is an important factor, but we restrict ourselves with composition.

We parse the melody by individual note events rather than grouping by beats or bars for the purpose of obtaining an event-level granularity. Therefore there is no preservation of beat or measure information. For example, if there are two eighth-notes in a beat, we create an event for each note (note level) rather than one event with two notes (beat level). The disadvantage is that metric hierarchy is lost but, on the other hand, this will make evident certain biases that would otherwise be masked. For example, syncopation introduced by the model will be amplified by parsing at the note level. This was chosen since it correlates with the method used by MusiCOG, thus making all models more comparable.

Since MusiCOG is a cognitive model [15], it handles the parsing of MIDI input using principles of music perception and cognition which are not included in the other two models and also does not require some of the preliminary parsing described above.

3.2 Corpus

The corpora consist of monophonic MIDI files from Classical, Jazz, Popular music, and Folk songs. These classes were selected for the purpose of investigating model behaviour in contrasting musical settings. We use a Finnish folk song collection that is available for research purposes [10], and manually created the other corpora by extracting melodies from MIDI files freely available on the web, each corpora adding up to roughly 7000 notes each. We settled with this number since the ratio of samples per transition is 30, for example, the 100 piece folk corpus consists of 24 interval types (12 ascending and 12 descending) and 10 time interval types: $7000/240 = 29.2$. For matters of space we present an analysis on the Folk corpus alone where we selected 100 pieces of 16 bars in length (**revise this**). The other corpora are around 30 pieces each and the compete collection is available for download together with analysis results [17].

4. OVERVIEW OF THE MODELS

4.1 Markov Models

Markov Chains are a widely used statistical approach. Two well known real-time systems implementing these techniques are Zicarelli's "M" [24] and Pachet's "Continuator" [20]. The theoretical basis lies in the field of *stochastics* which studies the description of random sequences dependent on a time parameter t. In their most basic form Markov Chains describe processes where the probability of a future event X_{t+1} depends on the current state X_t and not on previous events. In this way the sequences of notes in music can be analyzed to obtain a set of probabilities which describe the transitions between states, in this case the transitions between musical events.

As described by Conklin [6], perhaps the most common from of generation from Markov models is the so-called "random walk," in which an event from the distribution is sampled at each time step, given the current state of the model. After each selection the state is updated to reflect the selection. The *memory* or *order* of the model is the number of previous states that is considered, and thus defines the order of the Markov Chain. We implement a Variable Order Markov Model (VOMM) with a variability of 1-4 events.

4.2 Factor Oracle

Since music can be represented in a symbolic and sequential manner, pattern-matching techniques can be useful for the learning and generation of pattern redundancy and variations, respectively. The Factor Oracle [1] is one example of a text and/or biological sequence search algorithm that has been applied to music. It is an acyclic automaton with a linear growth in number of transitions with regards to the input pattern, which has been utilized in string searches [3]. There exists a construction mechanism [1] allowing the alphabet to be grown sequentially and the complete structure incremented online, thus allowing for the search of string factors in real-time.

It is important to note that neither the Markov Model nor the Factor Oracle will ever generate a transition that is not in the corpus. Also, it is conceivable that knowledge of the formal properties of each model could be used to evaluate model performance. However, as the corpus grows in size, knowledge of the formal properties of the models alone is not of much aid in predicting their behaviour. Hence the need for an empirical evaluation.

4.3 MusiCOG

MusiCOG, created by Maxwell [15], models perceptual and cognitive processes, with a special focus on the formation of structural representations of music in memory. The architecture is designed for the learning and generation of musical material at various levels (pitch, interval and contour), with an emphasis on the interaction between short- and long- term memory systems during listening and composition. As a cognitive model, with a complex hierarchical memory structure, there are many possible ways to generate output from MusiCOG. For this study, in order to reflect a similar systematic approach to the FO and MM, and to avoid music theoretical or compositional heuristics, we selected a relatively simple stochastic approach, which attempts to balance the application of both top-down (i.e., structural) and bottom-up (i.e., event transition) information.

MusiCOG is a feedback system, capable of interpreting its own output and modifying its behaviour accordingly. As an online model, MusiCOG will normally learn from its own output, but an option to disable this behaviour has been added and applied to half of the generation examples used for all tests in the current study. This was done in order to bring MusiCOG closer in functionality to the MM and FO, without entirely negating important aspects of its design.

room
room
room

5. METHODOLOGY

The analysis first requires extracting features from the input and output melodies, selecting significant features, and calculating similarity measures. Then, k-means and t-tests are used for clustering, calculating confusion matrices, and determining significant differences. Finally, we use Classic Multi-Dimensional Scaling (CMDS) for further investigating and interpreting the differences found. Figure 1 depicts a diagram outlining the methodology proposed.

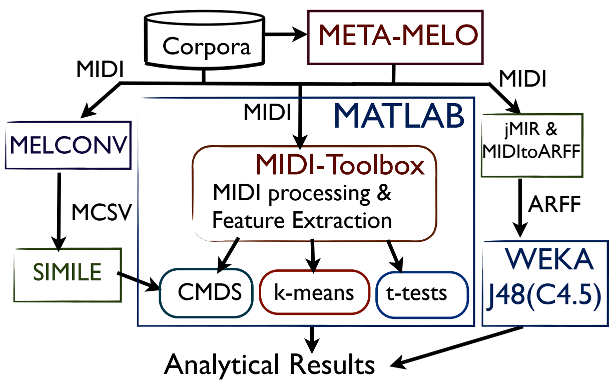

Figure 1. The proposed methodology, META-MELO is the generative component which is independent from the methodology.

MATLAB is used for most data processing, feature extraction, CMDS and t-test calculations. *SIMILE* [11] and *MELCONV* are Windows command line programs developed by Frieler for the conversion and comparison of MIDI monophonic files. The *Matlab MIDI Toolbox* [9] is used for a variety of functions. *WEKA* [12] is used for selecting the most significant features (C4.5 Java clone: J48) and for further data analysis and exploration. *MIDItoARFF* [23] and *jSymbolic* [16], a module of the jMIR toolbox from the same author, are also used for extracting features from MIDI files.

In Section 6.1 we describe the use of decision trees for selecting the features that best describe the differences in the models. These features are then used in Section 6.2 for visualizing the corpus and output of all models. In Section 6.3 and Section 6.4 we describe similarity analysis and CMDS respectively, used for evaluating the closeness of the groups of melodies. Finally, we performed pairwise, one-tailed t-tests for determining statistical significance, described in Section 6.5.

We trained each model with 100, 16 bar long pieces from the folk corpus and generated 32 pieces of 16 bars long from each model [17].

Intra and Inter-Model Melodic Similarity

Model	MM	MC	FO	CC
Markov (MM)	.206			
MusiCOG (MC)	.183	.208		
Factor Oracle (FO)	.166	.165	.198	
Folk Corpus (CC)	.178	.174	.154	.201

Table 1. Mean melodic similarity of model output and corpora using the "Opti3" similarity measure (1.0 = identity). Intra-model similarity is represented in the diagonal, lower value indicates higher diversity.

6. RESULTS

6.1 Decision Trees

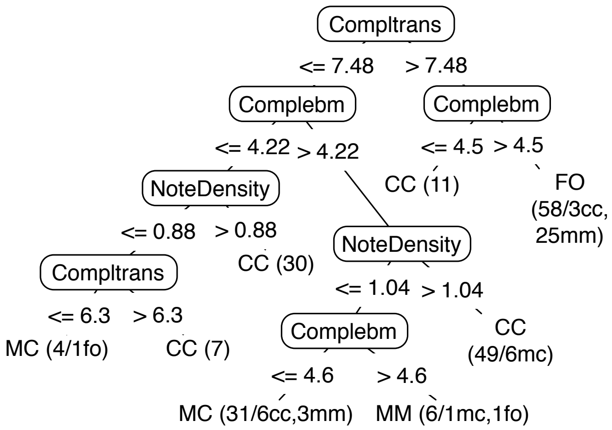

Figure 2. C4.5 decision tree (J48).

We used WEKA [12] for generating a C4.5 decision tree and reducing our feature space. This method was useful for arriving at an initial result in the search for musical biases by detecting which features, amongst the many extracted, distinguish the different models from the others or the corpus. Figure 2 shows a tree learned on the output from the Folk training and the corpus collection (CC). The three features arrived at by using the C4.5 decision tree are: (1) *Compltrans*, a melodic originality measure which scores melodies based on a 2nd order pitch-class distribution obtained from a set of 15 thousand classical themes. The value is scaled between 0 and 10 where a higher number indicates greater originality. (2) *Complebm*, an expectancy-based model of melodic complexity which measures the complexity of melodies based on pitch and rhythm components calibrated with the Essen collection. The mean value is 5 and the standard deviation is 1, the higher the value the greater the complexity of the melody. (3) *Notedensity*, the number of notes per beat. Details of these features can be found in the *MIDI Toolbox* [9].

The first number in the leaf is the count of instances that arrive to that leaf, the number after the dash, if present, indicates the count of those instances which are misclassified along with the correct class. Three aspects of the tree stand out: (1) Most of Markov output (MM), 25 melodies, are classified as Factor Oracle (FO) and are therefore not easy to distinguish. (2) The root (*Compltrans*) successfully separates 86% of FO and MM instances from 89% of CC and 100% of MusiCOG (MC), an indication of greater similarity between CC and MC. (3) The *Notedensity* feature seems to greatly aid in classifying and distinguishing MC from CC where other features are less successful. This type of analysis provides valuable diagnostic insight on the MC model since we can deduce that an increase in note density on the output would potentially improve the imitation capabilities of the model.

6.2 Originality and Complexity

In Figure 3 the corpus and the output instances for all models are plotted using the *Compltrans* and *Complebm* features described in Section 6.1. The plot shows a clear overlap between the corpus and MusiCOG, whereas Markov and Factor Oracle cluster together with higher values on both dimensions.

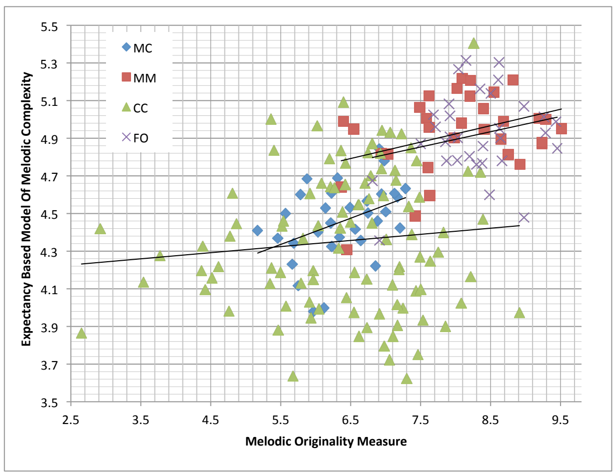

Figure 3. Expectancy Based Complexity and Originality. Folk corpus and model output, the corpus collection is marked as 'CC'.

6.3 Similarity Analysis

It has been noted by Müllensiefen and Frieler [18, 19] that hybrid measures have a greater predictive power than single-attribute measures for estimating melodic similarity. They provide an optimized metric 'Opti3,' which has been shown to be comparable to human similarity judgements [19]. Opti3 is based on a weighted linear combination of interval-based pitch, rhythmic, and harmonic similarity estimates, normalized between 0-1, where a value of 1 indicates that melodies are identical.

The corpus and the three sets of model output were analyzed to establish the similarity between them (inter-corpus analysis), as well as the diversity within the sets (intra-corpus analysis). We use the Opti3 measure and calculate the mean of the distances between each melody from the set in the row against all melodies in the column set (cartesian product). Looking at Table 1, we can interpret the diagonal as intra-model dissimilarity, the diversity of each set. Since a low value indicates higher diversity, MC is slightly the least and FO the most diverse of all sets, including the corpus. Furthermore, with this analysis, MM produces the output which is most similar to the corpus. This is the first discrepancy that we observe in the results.

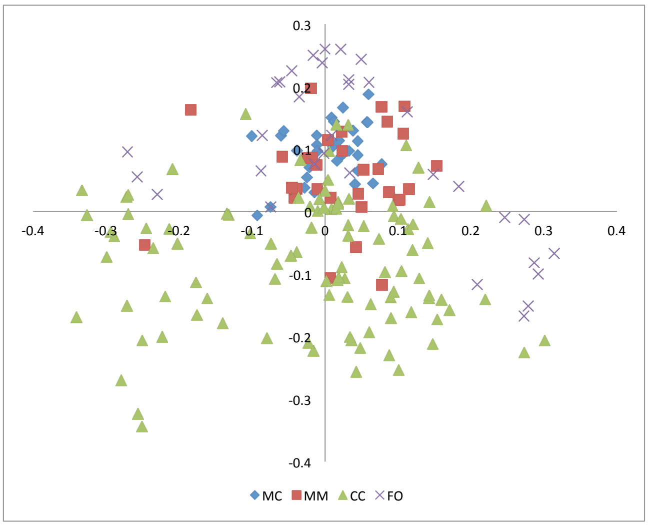

Figure 4. Multi-dimensional scaling for all models and corpus, using the optimized distance metric 'Opti3'. The corpus collection is marked as 'CC'.

6.4 Classic Multi-dimensional Scaling

CMDS is a multivariate analysis process similar to Principal Component Analysis (PCA), used for visualizing the clustering of N-dimensional data. We calculated dissimilarity matrices from the similarity measures obtained with "Opti3". In Figure 4 we can see that the horizontal axis separates quite generally the corpus from the model output. Although the dimensions are not easy to interpret, it is evident that the models do not explore the full 'creative' range of the corpus. Correlating with the similarity analysis, the diversity of FO output is apparent as it occupies a broader range in the space. It is worth noting that a similar topology was observed when scaling a set of 100 melodies from each model [17].

t-test Results

	Markov	MusiCOG	Factor Oracle
MusiCOG	IVdist	-	-
Factor Oracle	No	PCdist	-
Folk Corpus	PCdist	No	PCdist

Table 2. t-test results for the Folk Corpus, 'No' is indicated where no significance was found, otherwise the dimension where significance exists (P value < 0.0001).

6.5 Significance Tests

Table 2 shows the pairwise tailed t-tests that were performed on the 6 pairs of groupings of corpus and models for determining difference across four dimensions: Pitch-class distributions (PCdist), interval distribution (IVdist), contour and rhythm (*meldistance* function [9]). First, as in the similarity analysis, the mean distance between all melodies in one set is measured. Second, the distance from each melody in this set is measured against all of the melodies in the set with which it is being compared. Finally, the t-test is run on these two sets of measurements. Further details are available [17]). Where we found significance, it was at most in one dimensions, either PCdist or IVdist. In those cases the P value is < 0.0001. We clearly see that MusiCOG is different from both other models but is not differentiated form the Corpus. On the other hand, both Markov and the Factor Oracle are different from the corpus and undifferentiated between themselves. These results corroborate what is graphically displayed in Figure 3.

7. CONCLUSION AND FUTURE WORK

Returning to our broad definition of stylistic imitation, we expect successful models to roughly covering the same space as the corpus. The CMDS diagram shows graphically that this is not occurring in our study. Although this is an abstract conclusion, it is a more general and valuable one. This study also shows that the problem of stylistic imitation warrants further research.

We have also shown that the task of investigating for significance in the differences of the output is valuable for validating closeness to the corpus. The decision trees inform us, in musical features, as to what might be other important differences.

MusiCOG is a larger model and has more music domain knowledge, as it is informed by music perception as well as cognitive science and cognition. On the other hand, the VOMM and Factor Oracle models have no musical knowledge. This could be seen as a 'knowledge bias', which makes MusiCOG more true to the corpus. As such, this suggests continued investigation into developing musical cognitive models.

We leave the following for future work: the application of the methodology to polyphonic music, an in-depth analysis of the output of the models when trained on different corpora, and an evaluation of the behaviour of the models when combining stylistically diverse corpora (combinatorial creativity).

room
room

8. ACKNOWLEDGEMENTS

We want to thank Klaus Frieler for providing us his insights and software for melodic analysis as also the reviewers for the valuable comments. This research was made possible by a grant from the Natural Sciences and Engineering Research Council of Canada (NSERC).

9. REFERENCES

[1] C. Allauzen, M. Crochemore, and M. Raffinot. Factor oracle: A new structure for pattern matching. *SOFSEM'99: Theory and Practice of Informatics*, page 758, 2009.

[2] S. Argamon, S. Dubnov, and K. Burns. *The Structure of Style: Algorithmic Approaches to Understanding Manner and Meaning*. Springer-Verlag Berlin Heidelberg, 2010.

[3] G. Assayag and S. Dubnov. Using factor oracles for machine improvisation. *Soft Computing - A Fusion of Foundations, Methodologies and Applications*, 8(9):604–610, September 2004.

[4] R. Begleiter, R. El-Yaniv, and G Yona. On prediction using variable order markov models. *Journal of Artificial Intelligence Research*, 22:385–421, 2004.

[5] M. A. Boden. Computer models of creativity. *AI Magazine*, 30(3):23, 2009.

[6] D. Conklin. Music generation from statistical models. In *Proceedings Of The AISB 2003 Symposium On Artificial Intelligence And Creativity In The Arts And Sciences*, pages 30–35, 2003.

[7] D. Conklin and I. H. Witten. Multiple viewpoint systems for music prediction. *Journal of New Music Research*, 24:51–73, 1995.

[8] A. Cont, S. Dubnov, and G. Assayag. A framework for anticipatory machine improvisation and style imitation. In *Anticipatory Behavior in Adaptive Learning Systems (ABiALS)*. Rome, Italy, September 2006.

[9] T. Eerola and P. Toiviainen. *MIDI Toolbox: MATLAB Tools for Music Research*. University of Jyväskylä, Jyväskylä, Finland, 2004.

[10] T. Eerola and P. Toiviainen. Suomen kansan esavelmat. suomalaisten kansansavelmien elektroninen tietovaranto [digital archive of finnish folk tunes]. 2004.

[11] K. Frieler and Müllensiefen. The simile algorithms documentation. Technical Report. 2006. Available online at `http://doc.gold.ac.uk/isms/mmm/SIMILE_algo_docs_0.3.pdf`. Last visited on July 2013.

[12] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. The weka data mining software: an update. *SIGKDD Explor. Newsl.*, 11(1):10–18, November 2009.

[13] B. Manaris, P. Roos, P. Machado, D. Krehbiel, L. Pellicoro, and J. Romero. A corpus-based hybrid approach to music analysis and composition. In *Proceedings of the Twenty-Second AAAI Conference on Artificial Intelligence*, pages 839–845. AAAI Press.

[14] J. B. Maxwell, A. Eigenfeldt, and P. Pasquier. Manuscore: Music notation-based computer assisted composition. In *Proceedings of the International Computer Music Conference, In Ljubljana, Slovenia*, 2012.

[15] J. B. Maxwell, A. Eigenfeldt, P. Pasquier, and N. Gonzalez Thomas. Musicog: A cognitive architecture for music learning and generation. *Proceedings of the 9th Sound and Music Computing conference (SMC 2012)*, pages 521–528, jul 2012.

[16] C. Mckay and I. Fujinaga. jsymbolic: A feature extractor for midi files. In *In Int. Computer Music Conf*, pages 302–305, 2006.

[17] META-MELO. Online resource available at `http://metacreation.net/gonzalezthomas/mm/index.html`. Last visited on July 2013.

[18] D. Müllensiefen and K. Frieler. Cognitive adequacy in the measurement of melodic similarity: Algorithmic vs. human judgments. *Computing in Musicology*, 13(2003):147–176, 2004.

[19] D. Müllensiefen and K. Frieler. Melodic similarity: Approaches and applications. In *Proceedings of the 8th International Conference on Music Perception and Cognition*, 2004.

[20] F. Pachet. The continuator: Musical interaction with style. *Journal of New Music Research*, 32(3):333–341, 2003.

[21] P. Pasquier and A. Eigenfeldt. Proceedings of the first international workshop on musical metacreation. In *Conjunction with the The Eighth Annual AAAI Conference on Artifcial Intelligence and Interactive Digital Entertainment. AAAI Technical Report WS-12-16.*, page 88. AAAI Press, 2012.

[22] M. Pearce and G. Wiggins. Evaluating cognitive models of musical composition. In *Proceedings of the 4th International Joint Workshop on Computational Creativity*, pages 73–80, 2007.

[23] D. Rizo, P. J. Ponce de León, C. Pérez-Sancho, A. Pertusa, and J. M. Iñesta. A pattern recognition approach for melody track selection in midi files. In *Proc. of the 7th Int. Symp. on Music Information Retrieval ISMIR 2006*, pages 61–66, 2006.

[24] D. Zicarelli. M and jam factory. *Computer Music Journal*, 11(4):13–29, 1987.