Can Statistical Language Models be used for the Analysis of Harmonic Progressions?

Matthias Mauch*[1], Daniel Müllensiefen#[2], Simon Dixon*, Geraint Wiggins#

*Centre for Digital Music, Queen Mary, Univ. of London,
#Department of Computing, Goldsmiths, Univ. of London
[1]matthias.mauch@elec.qmul.ac.uk, [2]d.mullensiefen@gold.ac.uk

ABSTRACT

The availability of large, electronically encoded text corpora and the use of computers in recent decades have made Natural Language Processing (NLP) a flourishing research area. A wealth of standard techniques has been developed to serve use cases like document retrieval, identification of a finite vocabulary and synonyms, and the collocation of terms. Similarly, social networking among musicians in internet forums and the advent of automatic chord extraction have led to the establishment of chord databases, if on a smaller scale. Comparatively little research has been carried out on these growing corpora of chords. We suspect that one reason for this lack of research lies in the difficulty to decide if chords or other harmonic elements can be treated like lexemes in a text corpus. More simply, the question is: What is a word in terms of harmony? In this paper we propose a bottom-up approach. In order to find harmonic units whose distributions resemble distributions of words we consider chord elements differing in (a) length of chord sequence (counted in chord symbols), and (b) chord alphabet. Using lengths from 1 to 4 and two different chord alphabets we obtain a parameter space of size 8. For each of the parameter settings we compute statistical summaries of the resulting frequency distribution of the harmonic unit. As results, we report the parameter settings for two different chord corpora (2500+ songs each) that generate a frequency model corresponding most closely to the Brown Corpus, a general text corpus of American English.

I INTRODUCTION

Music and language may be processed independently and in different parts of the brain, but some obvious analogies relate the two domains, including evolution over time, and the auditory system as the primary gateway to perception. While there can be multiple independent melodies at a time in a piece of music, chords have the particular property that positions them close to language: they are perceived sequentially. In fact, for most tonal music we can assume that at each time point in a piece there is exactly one chord. Aptly, it is common to refer to sequences of chords as *chord progressions*.

The sequential nature of chord progressions has driven music computing researchers to use processing techniques known from text and speech processing tasks. For example, hidden Markov models similar to those used in speech recognition have been successfully incorporated into automatic audio chord labelling algorithms in order to achieve a smooth output (Bello and Pickens, 2005; Lee and Slaney, 2007). However, the models are still much simpler than their ancestors in speech recognition. Little attention has been paid to the problem of deciding *what* to model: most chord labelling algorithms assume that the nature of a chord is entirely determined by its intervalic content, with no reference to its duration or its metrical position, i.e. ignoring harmonic rhythm. In addition to that, it is (at least to us) very unclear whether just one chord is the harmonic unit to model and work with or whether progressions of two or even more chords form a basic harmonic unit which, for example, should be modelled as one state in a hidden Markov model. From music theory, it is known that one just a chord on its own (i.e. a root note and a chord type, e.g. d-minor) does not have a single function or meaning. Only in combination with the intervals to surrounding chords, in relation to a current tonal centre, and in connection with the corresponding temporal data, distinctive harmonic information emerges.

In NLP many powerful techniques have been developed in recent years to solve tasks in text processing that have close analogues in music processing. Among these tasks are the clustering of documents (or pieces in the case of music) according to genre and style, the retrieval of items from large databases according to similar content (query-by-example paradigm), the retrieval of identical structures despite differing surfaces (e.g. for detection of plagiarism or cover songs), the induction of (latent) syntactic rules, or identification of word collocations or idioms, i.e. lexemes that are commonly associated. Many successful technical applications like internet search engines, speech recognition systems or electronic dictionaries incorporate solutions to these tasks. The techniques developed to tackle these problems include Latent Semantic Analysis, n-gram, hidden Markov and other probabilistic models, statistics for word collocations, and probabilistic parsing and grammar approaches. All these techniques operate on tokens or basic units into which a text or an utterance can be split. In many linguistic studies or applications, words are used as the basic token and the statis-

tical techniques are constructed or optimised to work on large corpora of texts that have words as their basic token (unit). From there follows the rationale of this study: If we want to employ existing techniques from NLP to tackle analogue problems in music research using harmonic data (e.g. clustering according to style or cover song identification) we would like to find a representation derived from a large corpus of raw harmonic data that has comparable distributional properties like those that are typically found in verbal corpora. In other words, out of the many possibilities of what could constitute a "harmonic word" (i.e. the basic unit used for modelling) we aim at finding the one that generates a distribution that we can model using standard methodology from computational linguistics and that results in a model similar to those from linguistic corpora. We have to stress that we consider the analogy to the concept 'lexeme' or 'word' in language only in a distributional and maybe in a syntactical sense. We do, however, by no means imply that a 'harmonic word' has any semantic quality comparable to the verbal unit.

II TERMINOLOGY AND DATA

Two of the most important concepts of corpus linguistics are *type* and *token*. A type is a member of the dictionary of a language, typically a word (or morpheme), whereas a token is a member of the sample, the instance of a type. For example, in an English language corpus the type "that" could describe 612875 tokens. As we deal with very different kinds of data, one has to bear in mind that a type in a language corpus will naturally be a word (or word form) whereas in a corpus of harmonic data it will be some unit of chord information, be it a chord or a chord sequence (see Section III for details). In fact, the aim of this study is actually to find out what a good type representation for harmonic data might be.

A. Corpora

We use two different popular music corpora containing harmonic (chord) information, which we call *Community Corpus* and *Automatic Corpus*. Each set features more than $2,500$ songs.

The *Community Corpus* (2548 songs, 195874 chords) stems from several sources and has been compiled by numerous anonymous (amateur) musicians using the commercial software *Band in a Box* [1] . The main purpose of the program is to generate a MIDI accompaniment (in different styles) while taking as an input only a chord sequence (and optional melodies) provided by the user. From these Band in a Box files we extracted only the chord progressions, including metric duration. It is impossible to test thousands of songs for transcription accuracy, but we assume that very bad quality files are rare in spite of the likely lack of professional musical skill because *Band in a Box* plays the generated transcriptions and users can check their files by ear. The *Community Corpus* contains a wide range of songs, including many Jazz standards, but also classic popular songs and folk songs.

The chord data of the *Automatic Corpus* (2592 songs, 294264 chords) set was automatically extracted from polyphonic MIDI files using the algorithm proposed by Rhodes et al. (2007). The MIDI files used for this study was a sample from a professionally assembled collection of 14,063 songs acquired from the commercial MIDI distributor *Geerdes MIDI Music* [2] designed for professional use and karaoke playback. While the raw MIDI data is accurate, the automatic extraction is likely to be noisy. This set also is very diverse but—as the karaoke source suggests—features more contemporary and commercial pop music.

We compared these two music corpora to the *Brown Corpus*, a standard and widely researched text corpus of written American English published in 1967 which consist of 500 samples from different contemporary text genres [3] (45,215 different words or types and 1,006,770 tokens or words in total).

B. Data Format, Chord Classes

The *Automatic Corpus* assigns to every beat (retrieved from the MIDI representation) a chord label consisting of the chord root as well as the chord "quality" chosen from a set of six labels `maj`, `min`, `dim`, `aug`, `sus9`, `sus4`. We adopt this format for the (originally richer) *Community Corpus* and map the chords appearing in it to these six classes [4] . Chords that do not fit with any of the six classes are assigned to an auxiliary class called *unknown*. Both chord data sets are stored in RDF files using the Chord Ontology (Sutton et al., 2007), making them the two largest chord corpora in open format we are aware of. We plan to publish the *Community Corpus* and the *Automatic Corpus* later this year on the website `www.chordtranscriptions.net`.

III METHOD

The analytical characterisation of the two pop corpora in comparison with the *Brown Corpus* makes extensive use of the zipfR package (Evert and Baroni, 2007) for lexical statistics within the R programming language. We largely follow the analytical procedures proposed in the tutorial introduction of the package [5] .

While the text corpus data is already provided by the zipfR package, we have to compile statistics for the chord corpora from the RDF files (see B.). In order to do so we load each chord corpus into our software and generate a suffix tree (for general information on suffix trees, see (Gusfield, 1997)). The suffix tree structure allows us to conveniently access much of the information we need for further processing. We consider different "alphabets" of

[1] http://www.band-in-a-box.com/

[2] http://www.midimusic.de/

[3] http://khnt.aksis.uib.no/icame/manuals/brown/

[4] for a table see
http://chordtranscriptions.net/ChordLists/chordlist_reduced.csv

[5] http://www.cogsci.uni-osnabrueck.de/~severt/zipfR/materials/zipfr-tutorial.pdf

harmonic elements or harmonic units which change with the parameters described in this section.

A. Parameter Space

While in texts words often have boundaries marked by space characters and the like, the chord sequence of a song provides us only with the information of chord changes. We look at four different kinds of harmonic elements, namely at single chords, and chord progressions of length 2, 3, and 4. Please note that only the first option is non-overlapping. We represent only the chord class of each chord (six different classes, see above) and the interval between successive chords. We assume that any song can be transposed into any key and still remains the same. In much the same way as Mauch et al. (2007) we transpose any of the items presented above so that the first chord has root C. Hence, two chord progressions such as C`maj` – F`min` – G`maj` and D`maj` – G`min` – A`maj` will be considered equivalent. Behind this procedure is the belief that the key of a piece is a concept that facilitates composition and performance but is less important for the listener (except for individuals with absolute pitch). Also, as reliable key information is not at hand, this is a convenient way to implement transposition invariance.

As we want to investigate the influence of harmonic rhythm, we included duration information, as both collections of songs contain information on the metric duration (quantised to beats) of the chords. In order to make the information manageable, we mapped the durations into three duration classes, namely `1beat`, `23beats`, and `manybeats`, where `1beat` captures all chords that have a duration of one beat, `23beats` those with duration of either 2 or 3 beats and `manybeats` those with 4 or more beats. For example, if one distinguishes chords with different harmonic duration,

$$\text{C}\texttt{maj-1beat} \longrightarrow \text{F}\texttt{min-manybeats}$$

differs from

$$\text{C}\texttt{maj-manybeats} \longrightarrow \text{F}\texttt{min-manybeats},$$

while they obviously would not differ if one considered only the chord quality and root differences. The possible parameter settings are hence to use metric durations or not to (and hence assume all chord durations are equal).

Taken together, the 4 different length options for the chord progression combined with the two types of alphabet generate 8 different parameter settings that we explore in this study. Future investigations following this approach should also test the significance of metrical position of a chord in a bar, duration ratio between successive chords, the relation to the current key, and different chord class sets. This will, of course, increase the number of parameter settings.

B. Type Rankings

The most straightforward statistic of a text corpus is the ranking of types. The types we are considering are chord n-grams on the one hand and words on the other. This representation has a value in its own right (Mauch et al., 2007) and can certainly serve to get an overview of the (trivial) characteristics of a corpus (Table 1). To illustrate how the harmonic alphabet and the resulting distribution of relative frequencies differs between two parameter settings, we picked two parameter settings at random, chord sequences of length 1 including metric duration and chord sequences of length 4 ignoring duration information, and list them along the relative frequencies of words in the *Brown Corpus*.

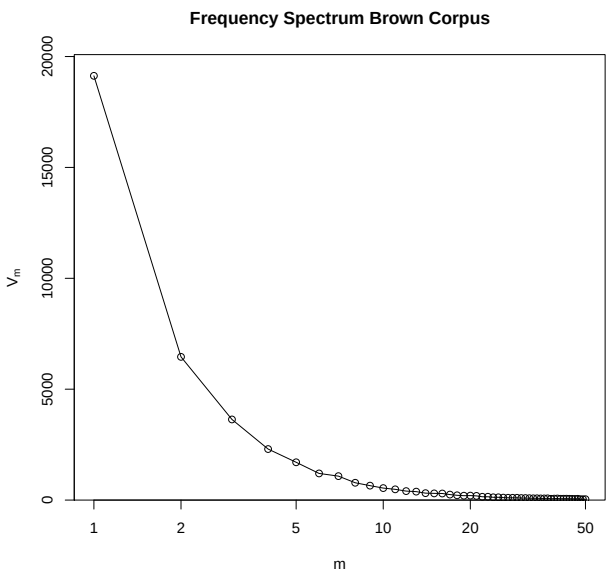

Figure 1. Word type frequency spectrum of the *Brown Corpus*. m is the frequency class, and V_m the corresponding number of word types. For example, the number of different words (word types) that occur $m = 2$ times in the corpus is about $V_2 = 6500$.

C. Frequency Spectra

The frequency spectra of linguistic corpora are characterised by the fact that many words occur only once or very rarely in the corpus while only a few words are used very frequently. The frequency spectrum from the *Brown Corpus* illustrates this nicely where we see the the frequency class $m = 1$ being by far the class that includes the largest number of different words (V_m), i.e. types.

In Figure 2 one can see an empirical frequency spectrum for chord progressions of length 3 in combination with metric duration. The frequency spectrum resembles the spectrum obtained from the *Brown Corpus* to a certain degree.

We obtain a completely different spectrum from the *Automatic Corpus* corpus when computed from a representation of single chords (length=1) in combination with the metrical duration information (see Figure 3): None of the harmonic types appears only once and types occurring in the corpus have a much higher frequency, i.e. are re-

Freq. rank	— *Community Corpus* — metric durations, length 1		no durations, length 4		*Brown Corpus*	
	rel. freq	type	rel. freq	type	rel. freq	type
1	0.309	maj-manybeats	0.022	$\texttt{maj} \xrightarrow{5} \texttt{maj} \xrightarrow{7} \texttt{maj} \xrightarrow{5} \texttt{maj}$	0.069	the
2	0.264	maj-23beats	0.016	$\texttt{maj} \xrightarrow{7} \texttt{maj} \xrightarrow{5} \texttt{maj} \xrightarrow{7} \texttt{maj}$	0.036	of
3	0.128	min-23beats	0.014	$\texttt{maj} \xrightarrow{5} \texttt{maj} \xrightarrow{5} \texttt{maj} \xrightarrow{7} \texttt{maj}$	0.028	and
4	0.117	min-manybeats	0.012	$\texttt{min} \xrightarrow{5} \texttt{maj} \xrightarrow{5} \texttt{maj} \xrightarrow{5} \texttt{maj}$	0.026	to
5	0.061	maj-1beat	0.011	$\texttt{min} \xrightarrow{5} \texttt{maj} \xrightarrow{7} \texttt{min} \xrightarrow{5} \texttt{maj}$	0.023	a
6	0.031	dim-23beats	0.011	$\texttt{maj} \xrightarrow{5} \texttt{maj} \xrightarrow{5} \texttt{maj} \xrightarrow{5} \texttt{maj}$	0.021	in
7	0.022	min-1beat	0.009	$\texttt{maj} \xrightarrow{5} \texttt{min} \xrightarrow{5} \texttt{maj} \xrightarrow{5} \texttt{maj}$	0.011	that
20	< 0.001	sus9-23beats	0.006	$\texttt{maj} \xrightarrow{5} \texttt{maj} \xrightarrow{5} \texttt{maj} \xrightarrow{2} \texttt{maj}$	0.005	at
50		—	0.002	$\texttt{min} \xrightarrow{5} \texttt{maj} \xrightarrow{5} \texttt{maj} \xrightarrow{9} \texttt{maj}$	0.002	if
100		—	0.001	$\texttt{maj} \xrightarrow{5} \texttt{maj} \xrightarrow{5} \texttt{maj} \xrightarrow{11} \texttt{maj}$	0.001	way
200		—	0.001	$\texttt{maj} \xrightarrow{2} \texttt{min} \xrightarrow{8} \texttt{maj} \xrightarrow{2} \texttt{maj}$	< 0.001	hand
1000		—	< 0.001	$\texttt{min} \xrightarrow{1} \texttt{min} \xrightarrow{5} \texttt{maj} \xrightarrow{6} \texttt{min}$	< 0.001	charles
10000		—	< 0.001	$\texttt{min} \xrightarrow{11} \texttt{dim} \xrightarrow{7} \texttt{min} \xrightarrow{5} \texttt{maj}$	< 0.001	registry

Table 1. Type rankings and relative frequencies for two selected parameter settings in the *Community Corpus* as well as the *Brown Corpus*. For the chord sequences of length 4, the root difference between to consecutive chords is represented as an upwards interval measured in semitones, i.e. the chord change C-F would be an instance of $\texttt{maj} \xrightarrow{5} \texttt{maj}$.

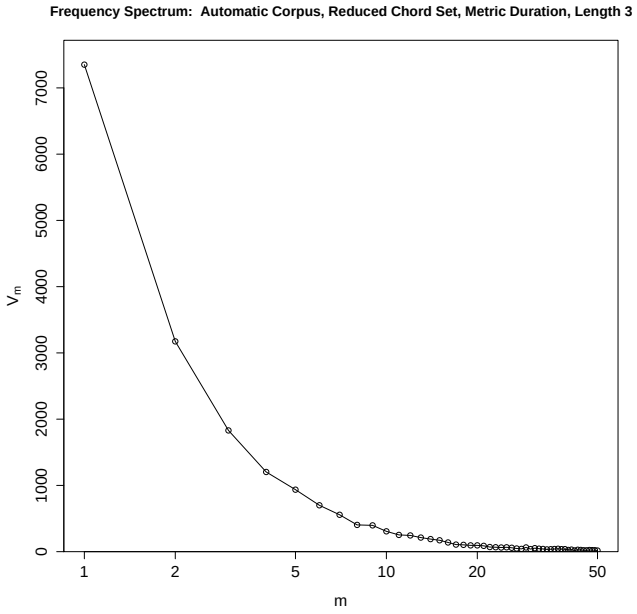

Figure 2. Harmonic type frequency spectrum of *Automatic Corpus* using chord sequences of length 3 and metrical duration information. Similar in shape to Figure 1.

peated much more often.

D. Productivity

A common way to summarise these frequency spectra is to divide the number of words which only occur once (V_1, technical term: hapax legomena) by the overall number of tokens in the corpus (N). The quotient is simply the proportion of types that have exactly one instance in the sample. It gives an indication of how the vocabulary is used and how productive the process is that generated the corpus, hence it is often called *measure of productivity*. Table 2 lists the productivity values for all parameter settings in our study. Baayen (1994) explains in more detail the use of the productivity measure in language.

E. LNRE modelling

The characteristic shape of the frequency spectra arising from linguistic corpora can be modelled by so-called *Large Number of Rare Events models* that allow us to summarise the frequency distribution from a corpus by a few model parameters. Out of the several different models applicable for this type of distribution we chose the finite Zipf-Mandelbrot model as described by Evert (2004):

$$g(\pi) = \begin{cases} C \cdot \pi^{-\alpha-1} & A \leq \pi \leq B \\ 0 & \text{otherwise,} \end{cases} \qquad (1)$$

where $C = (1-\alpha)/(B^{1-\alpha} - A^{1-\alpha})$ is a normalising factor. What is modelled here is the density $g(\pi)$ of types depending on their probability π. A quick intuitive explanation could go as follows: In a finite corpus the probability of a type (estimated by its relative frequency) never

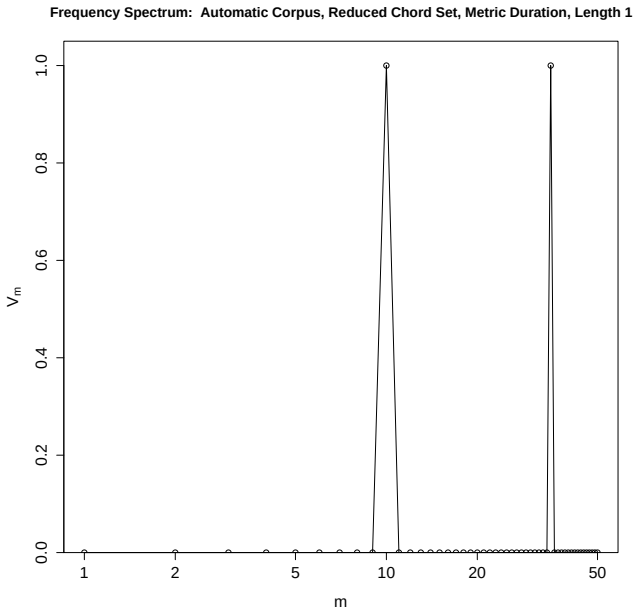

Figure 3. Harmonic type frequency spectrum of *Automatic Corpus* using chord sequences of length 1 and metrical duration information. One can see only a small portion of the data, as most types are concentrated in very high frequency classes, similar to what can be observed in Table 1.

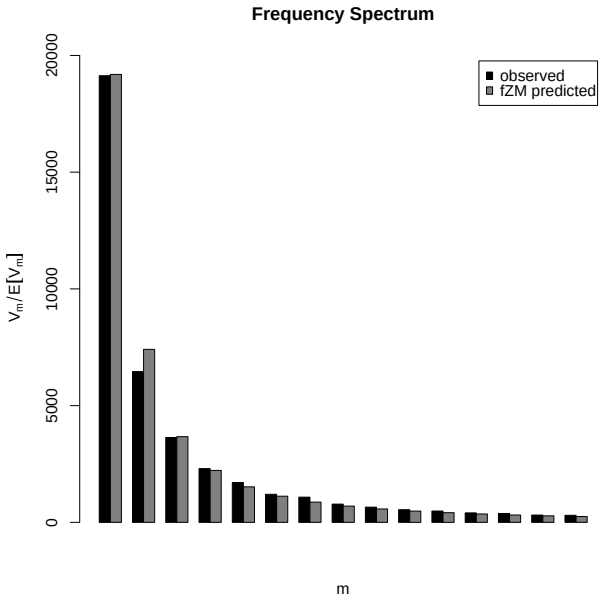

Figure 4. Observed and predicted frequency spectrum for the *Brown Corpus* from finite Zipf-Mandelbrot model

falls below some (small) positive number. The parameter A represents that number. Similarly, the most frequent type has a relative frequency somewhere between 0 and 1, and intuitively the density $g(\pi)$ should be zero for values of π greater than that. Hence the use of B. The most interesting parameter however is α: the basic assumption of the model is that there are many types which occur rarely, i.e. have a low probability. The density at values close to zero is therefore high, which is modelled by a power law, in which α characterises the slope of the type density curve. For a more formal derivation, see (Evert, 2004).

We fitted the finite Zipf-Mandelbrot model to the *Brown Corpus* and to all 16 frequency spectra resulting from parameters the eight parameter settings for each of the two corpora. We used the Simulated Annealing algorithm for the model fitting and parameter optimisation.

As visual examples, we present graphs of frequency spectra as observed and as predicted by the finite Zipf-Mandelbrot model for the *Brown Corpus* ($\alpha = 0.578$; Figure 4) and two parameter settings: *Community Corpus* without duration information and length 4 (Figure 5), and *Automatic Corpus* with metric duration and length 3 (Figure 6), as the latter two have α values (0.345 and 0.605) closest to that of the *Brown Corpus*. Looking at the observed frequency distributions and the corresponding model predictions for these parameter settings seems to suggest a reasonable good model fit without any clear pattern of deviations between observed and predicted numbers of the first 15 frequency ranks.

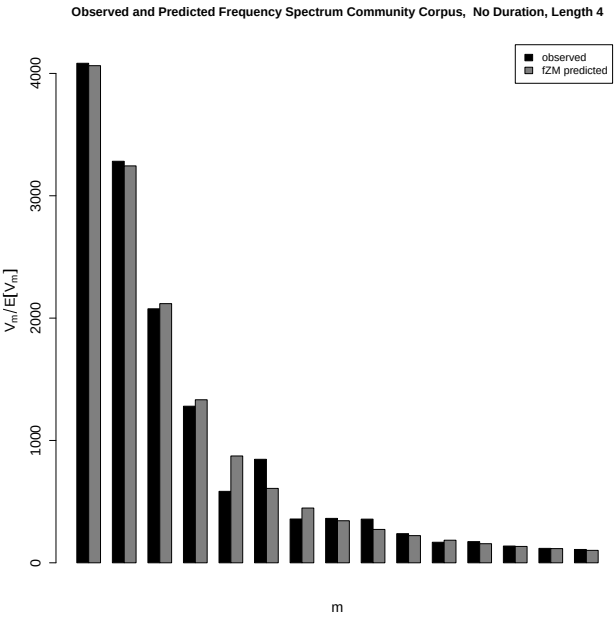

Figure 5. Observed and predicted frequency spectrum for the *Community Corpus* from finite Zipf-Mandelbrot model. Parameter setting: Length of chord progressions = 4, not using durational information.

	Corpus	Duration	Length	Exp. Voc. Size at 100k tokens	Max. Voc. Size	Productivity	alpha	B
1	Community	No Duration	1	7	(7)	0.0000	0.000	1.429
2		No Duration	2	380	(588)	0.0005	0.112	0.032
3		No Duration	3	3906	(49392)	0.0110	0.345	0.007
4		No Duration	4	12820	(4148928)	0.0498	0.857	0.008
5		Metric Duration	1	21	(21)	0.0000	0.000	0.419
6		Metric Duration	2	1657	(5292)	0.0032	0.173	0.008
7		Metric Duration	3	12122	(1333584)	0.0450	0.802	0.005
8		Metric Duration	4	27465	(336063168)	0.1297	1.000	0.005
9	Automatic	No Duration	1	5	(6)	0.0000	0.000	2.156
10		No Duration	2	190	(432)	0.0002	0.098	0.076
11		No Duration	3	2481	(31104)	0.0070	0.315	0.008
12		No Duration	4	12204	(2239488)	0.0602	0.650	0.004
13		Metric Duration	1	14	(18)	0.0000	0.012	0.773
14		Metric Duration	2	961	(3888)	0.0017	0.098	0.011
15		Metric Duration	3	13532	(839808)	0.0599	0.605	0.002
16		Metric Duration	4	38819	(181398528)	0.2563	0.983	0.002
17	Brown			12780	—	0.063	0.578	0.002

Table 2. Resulting bench mark values for all tested parameter settings in comparison with the *Brown Corpus*. The Maximum Vocabulary Size figures represent the theoretical vocabulary size possible by using the respective alphabet.

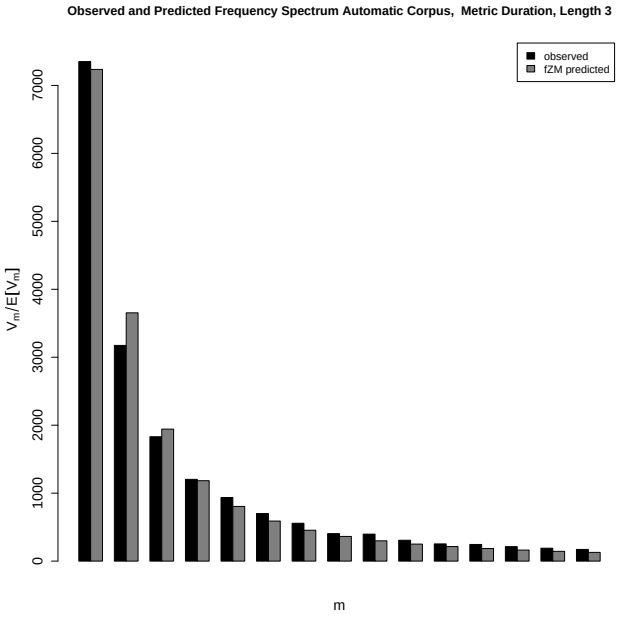

Figure 6. Observed and predicted frequency spectrum for the *Automatic Corpus* from finite Zipf-Mandelbrot model. Parameter setting: Length of chord progressions = 3, using metrical durations.

IV RESULTS

Table 2 contains the results for all the different parameter settings considered. We focus primarily on three bench mark values: The extrapolated expected vocabulary size at a corpus size of 100,000 tokens (i.e. for the *Brown Corpus* the vocabulary size after the first 100,000 tokens), the measure of productivity, and the α parameter from the *LNRE*- model. For each of these bench mark values we look for input parameter settings that generate values within a comparable range (i.e. roughly within the same order of magnitude) to the *Brown Corpus*. Taking this rather qualitative look at the results table, we find the closest approximation to the *Brown Corpus* for the metrical information setting of length 3 and the non-metrical setting of length 4. This is finding holds true for both the *Community Corpus* and the *Automatic Corpus* (see rows 2, 10 and 8, 16 of Table 2 respectively)

V DISCUSSION

We are aware that there are many more parameters that can be introduced into the current framework. Also, we certainly know that there are many more ways of looking at chords and harmonic units that do not start from a model of (overlapping) n-grams. For example, one of the most distinctive properties of music, repetition (Huron, 2006), has been excluded in this paper. But we would like to include in a future extension of this study repetition-based methods for the segmentation of a continuous symbol stream, like the algorithm proposed by Cohen and Adams (2002). This would allow to cut the stream of

chord symbols into units of variable lengths. In addition and following the present approach, one could also consider reducing all chords to their roots (i.e. using only 1 instead of 6 chord classes) or on the contrary, extending the number of chord classes. Another option includes using a fixed time window of, say, one or two bars, as a parameter of the present framework instead of a fixed length for the chord progressions. In addition to the present comparison with the word frequency models distribution, looking instead at distributions of part-of-speech tags might offer some more insight.

VI CONCLUSIONS

For this study we compiled and used the largest manually generated chord corpus we are aware of, as well as a chord corpus automatically extracted from MIDI data. We have identified two parameter settings for treating chord information that result in a type frequency spectra that seem to be comparable to a spectrum from a linguistic corpus and that can be modelled similarly by a LNRE model. One setting includes metrical duration and the other one makes no use of this durational information. We do not claim that this parameter setting is the optimal or most adequate one for a harmonic representation (this will be subject to further and more rigourous testing and exploration of the parameter space). But these settings allow us to look at the pop song corpora with a *resolution* comparable to the *Brown Corpus*. At this point it is not taken as granted that this resolution is adequate for investigation of the harmonic content of pop music corpora. Only when we use the resulting harmonic units in an application (e.g. LSA for musical style clustering or cover song detection) will we be able to test whether the generation of harmonic units actually has given useful and in that sense meaningful results. Nonetheless, the exploratory investigations presented in this paper are a necessary first step for pruning the enormous parameter space that is potentially relevant when we aim at finding a reliable unit for harmonic modelling.

References

R. Harald Baayen. Productivity in language production. *Language and Cognitive Processes*, 9(3):447–469, August 1994.

Juan P. Bello and Jeremy Pickens. A Robust Mid-level Representation for Harmonic Content in Music Signals. In *Proc. ISMIR 2005, London, UK*, 2005.

Paul Cohen and Niall Adams. An unsupervised algorithm for segmenting categorical timeseries into episodes. In *Pattern Detection and Discovery, ESF Exploratory Workshop, London, UK, September 16-19, 2002, Proceedings*, 2002.

Stefan Evert. A simple LNRE model for random character sequences. In *Proceedings of JADT 2004*, pages 411–422, 2004.

Stefan Evert and Marco Baroni. zipfR: Word frequency distributions in R. In *In Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics, Prague, Czech Republic*, 2007.

Dan Gusfield. *Algorithms on Strings, Trees, and Sequences: Computer Science and Computational Biology*. Cambridge University Press, 1997.

David Huron. *Sweet Anticipation: Music and the Psychology of Expectation*. MIT Press, 2006.

Kyogu Lee and Malcolm Slaney. A Unified System for Chord Transcription and Key Extraction Using Hidden Markov Models. In *Proceedings of the 2007 ISMIR Conference, Vienna, Austria*, 2007.

Matthias Mauch, Simon Dixon, Christopher Harte, Michael Casey, and Benjamin Fields. Discovering Chord Idioms through Beatles and Real Book Songs. In *ISMIR 2007 Conference Proceedings, Vienna, Austria*, 2007.

Christophe Rhodes, David Lewis, and Daniel Müllensiefen. Bayesian Model Selection for Harmonic Labelling. May 2007. MCM Berlin.

Christopher Sutton, Yves Raimond, Matthias Mauch, and Christopher Harte. The Chord Ontology, 2007. URL `http://purl.org/ontology/chord/`.