A STUDY ON N-GRAM INDEXING OF MUSICAL FEATURES

葉志立 *YIP Chi Lap* clyip@csis.hku.hk

Ben KAO kao@csis.hku.hk

Department of Computer Science and Information Systems, The University of Hong Kong

ABSTRACT

Since only simple symbol-based manipulations are needed, n-gram indexing is used for natural languages where syntactic or semantic analyses are often difficult. Music, whose automatic analysis for patterns such as motifs and phrases are difficult, inaccurate or computationally expensive, is thus similar to natural languages. The use of n-gram in music retrieval systems is thus a natural choice.

In this paper, we study a number of issues regarding n-gram indexing of musical features using simulated queries. They are: whether combinatorial explosion is a problem in n-gram indexing of musical features, the relative discrimination power of six different musical features, the value of n needed for them, and the average amount of false positives returned when n-grams are used to index music.

1. INTRODUCTION

Because of its simplicity, n-gram indexing has been used in text retrieval systems for years. It is done by associating every n-character fragment, called an n-gram, of a document with the document identifier in an inverted index. It is particularly suitable for indexing text in languages without easy-to-parse word boundaries, such as Chinese, Japanese, or Korean. Only simple character-based manipulation is needed to build n-gram indices; elaborate syntactic or semantic analysis of text, which are particularly difficult for these languages, are not required.

Music, seen as a temporal sequence of notes and rests, is similar to those natural languages in one aspect: although it is often easy to obtain note sequences from computer representations of music, further analysis for musical patterns such as motifs, phrases and movements are often difficult, inaccurate, or computationally expensive. This is because music, like natural languages, do not have rigid grammatical rules, and is multidimensional in the sense that the same piece can be interpreted in multiple ways [4]. Simple indexing techniques that do not require extensive musical analysis thus have their appeal for music indexing in retrieval systems, and n-gram indexing is thus a natural choice.

In indexing natural languages, bigrams and trigrams, that is, n-grams with $n = 2$ or 3, are most often used. Indeed, the use of bigrams and trigrams fits the characteristics of natural languages. For example, since more than two-third of Chinese words consist of two characters [5], bigram indexing is often used for Chinese. For the same reason, n is often 2 or 3 for Korean [3]. However, although n-gram indexing seems to be suitable for music, few studies can be found in the literatures on what the n should be for musical n-grams. Since there are potentially $|\Sigma|^n$ n-gram combinations for a musical feature sequence with $|\Sigma|$ alphabets, combinatorial explosion would be a problem if a large n is needed.

Besides the problem of potential combinatorial explosion, other issues arise when n-grams are used for indexing. Unlike indexing methods that use data structures for string matching such as the suffix tree [6], positional information are normally not stored in n-gram indices. Since a query is considered to match a document when all its n-grams can be found in it, false positives may occur. For example, if bigram indexing ($n = 2$) is used, the query `abcd` is considered to match the text `abacdbc` because all the query bigrams `ab`, `bc` and `cd` can be found in the text, though `abcd` cannot be found in `abacdbc`. False drops, on the other hand, are not possible.

To our knowledge, few articles directly addressed the issue of indexing musical features using n-grams. Two of them are [2] and [7]. In [2], 4- to 6-grams of musical intervals quantized in four ways were studied in an informetric framework. In [7], the average number of pieces musical n-grams can match was studied. In both works, statistics was based on the assumption that all n-grams were selected with equal probability. Since in real applications this assumption may not always hold, in this paper, we study the issues of an n-gram-based music retrieval system using simulated queries. In particular, we seek to answer the following questions. First, whether combinatorial explosion is a problem in musical n-gram indexing. Second, the value of n needed for six different musical features when the statistical distribution of n-gram is taken into account. Third, the relative discrimination power of different features, and fourth, the average number of false positives returned when n-grams are used to index music.

0-7803-6536-4/00/$10.00 (c) 2000 IEEE

2. THE MUSIC COLLECTION

Since the Internet is one of the major sources of digital music, our collection consists of 1003 MIDI files of songs and music popular in Hong Kong, Taiwan, or Japan, all obtained from the Internet. All the pieces were assumed to be in equal temperament tuning system. The MIDI tracks containing melodies were manually extracted for the experiments. Because some of the MIDI files contain performed rather than notated music, automatic onset quantization was done on all the pieces to reduce the effect of performance timing variabilities. Essentially, all note onset and offset times were quantized to fit into a set of time values determined by parameters such as the time signature of the piece. The sequence of notes and rests obtained were then analyzed for musical features.

3. MUSICAL FEATURES

Three traditionally used musical features, namely Interval Sequence, Profile and Note Duration Ratio Sequence, and three "Coarse Interval" features introduced in [7], namely CI3, CI5 and CI7, were used in the experiments. An Interval Sequence is a sequence of numbers each denoting the musical interval size, in number of semitones, between two temporally consecutive notes. Coarse Interval sequences CIx are uniformly quantized Interval Sequences where x intervals are grouped together with the unison interval at the center of a group. Mathematically, an interval of s is mapped to $sgn(s)\left\lfloor \frac{|s|+n}{2n+1} \right\rfloor$ for the feature CI$\{2n+1\}$. The feature Profile shows the shape of the melody by the ups and downs of its note pitches; it is nonuniformly quantized Interval Sequence using the three-valued sign function. Those features derived from Interval Sequence (i.e., CIx and Profile) often have a smaller alphabet size than Interval Sequence, which can help reducing index sizes. A Note Duration Ratio Sequence is a sequence of ratios of sounding durations between temporally consecutive notes. Some examples of the features introduced here can be found in [7].

4. THE EXPERIMENTS

In our experiments, every piece of music in our collection is first analyzed for the described features. Then, for every feature, multiple n-gram indices are built with different values of n. To search a particular n-gram index, a given query of length $q \geq n$ feature points is first broken down into $(q-n+1)$ n-grams. By looking up the n-gram index, we obtain, for each piece in the database, the number of query n-grams that matches it. A query is considered to completely match a piece when all the $q-n+1$ query n-grams can be found in it. In our experiments, the values of n from 1 to 8 are used. Results where less than $q-n+1$ query n-grams are matched cannot be reported here due to space limitations.

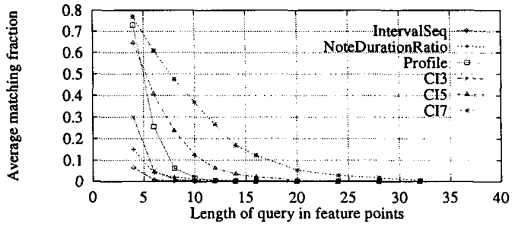


Figure 1: Average fraction of database containing matching pieces using string matching method

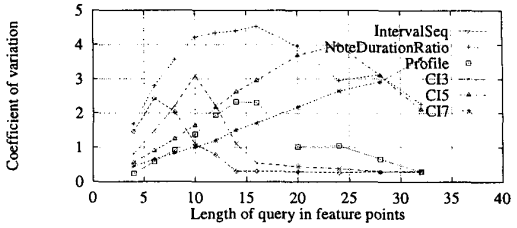


Figure 2: Coefficient of variation on number of matching pieces using string matching method

Queries in our experiments are generated by sampling methods. First, a query length of q feature points is fixed. Then, segments of musical features of length q is selected from the musical features of the pieces so that the probability of any segment q feature points in length being selected is the same. This way, for any given n-gram, its probability of being in the length-q segment will be proportional to its occurrence frequency in the collection. In our experiments, 1000 queries are generated for each value of q, and the value of q varies from 4 to 32 in our experiments.

To compare the number of matching pieces when n-gram indexing is used under different values of n, we match every set of length-q query samples with all the n-gram indices for all $n \leq q$. The queries are also matched against the database using an exact string-matching method. This gives us match results in which no false drops nor false positives can occur.

5. RESULT AND ANALYSIS

To study the relative discrimination power of the features, Figure 1 shows a graph on the average fraction of the database that matches the samples against the length of query in feature points using a string matching method. Also, curves of coefficient of variation, that is, standard deviation normalized by the average value, for different features are also plot in Figure 2. As expected, as the query

length q increases, the average fraction of songs in the database that match the queries decreases, since the queries become more specific. However, comparing the six curves, it is found that a rather large n is needed for CI5 and CI7 curves to be sufficiently low; on average, more than 10% of the database gives a match even when the queries consist of 16 CI7 feature points. In contrast, a smaller n is needed for the Profile curve. Since Profile is Interval Sequence nonuniformly quantized to three values, with the fact that musical intervals tend to be small-sized in the musical scale [1], this observation indicates that uniformly quantized Interval Sequence with large quantization step size are not discriminating features. Indeed, the unison interval has been found to be an important discriminant of music [7]. For features based on Interval Sequence to be more discriminating, finer quantization step size could be applied (e.g., use CI3 rather than CI7), or nonuniformly quantization can be done. Other possibilities that can be investigated in future research include use of Interval Sequence nonuniformly quantized to more than three levels.

The coefficient of variation shows the dispersion of data relative to the average. Note that the coefficient of variation for a feature is not very meaningful when n is so small that a large proportion of the database is matched, since the standard deviation cannot become too large from the average value. This is the case for most features when $q = 4$ and the coarse intervals when $q = 6$. From the graph, we found that the coefficients of variation for the features Interval Sequence and CI3 reach a value of below one at a moderate value of n around 15. That is, the relative dispersion of the fraction of matching pieces is small for such an n value. Since practical query lengths for Interval Sequences and CI3 are also at that range, these features are good in the sense that the number of matching pieces using two queries of the same length would not differ by too much; they "partition" the database more or less evenly. The feature Profile also shows a similar property although a higher value of n is needed. Coarse Interval features and Note Duration Ratio Sequences, on the other hand, give relatively high value of coefficient of variation in our considered range of query lengths. This means some of their feature sequences match very few pieces in the database, while some match a lot. In other words, some of these feature sequences occur in many pieces while some very few. These features can still be good features to use provided that feature sequences that match a lot of pieces can be identified and removed. This is similar to stopword removal in natural language text retrieval systems.

False positives can affect retrieval efficiency because piece-by-piece filtering is needed to remove them. To study the amount of false positives n-gram indexing would give, for every feature we found the average proportion of songs in the database that are false positives. The values found for every n considered are plot against the length of queries and are shown in Figure 3. Each graph in the figure corresponds to one feature, and each curve in the graphs corresponds to one value of n. The topmost curves are for $n = 1$, second topmost for $n = 2$, and so on.

A number of observations can be made from these graphs. For example, the false positive proportion for CI5 can be more than 10% unless n is larger than 7, and that for CI7 is always larger than 7.5% for query lengths more than 12 feature points even when $n = 8$. This means that relatively large values of n are required for them, which can offset the gain obtained by their smaller alphabet size compared to Interval Sequence. Further investigation is needed on whether there are improvements when, as previously described, feature stopwords are identified and removed.

From these graphs, we also found that the value of n is different for different features. For example, with a moderate query length of 12 feature points, to achieve a false positive proportion of less than 2%, the values of n required are at least 3, 5, 7 and 5 for Interval Sequence, CI3, Profile and Note Duration Ratio Sequence respectively. An n of more than 8 is required for features CI5 and CI7. Since these values of n for the first four features are not very large and in general features requiring a larger n have smaller alphabet sizes [7], combinatorial explosion would not be a serious problem in n-gram indexing of most musical features studied here. In short, n-gram indexing is a feasible scheme to use in music retrieval systems.

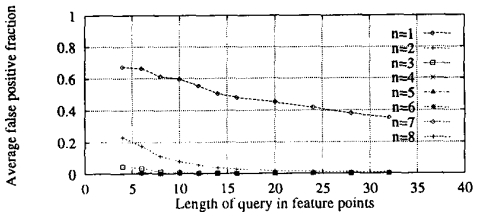
Note that the value of n observed above should be interpreted differently from those recommended in [7]. It is because in that paper, all n-grams were considered to be selected equally probably, while here, the sampling procedure automatically takes n-gram distribution into account.

6. SUMMARY AND FUTURE WORK

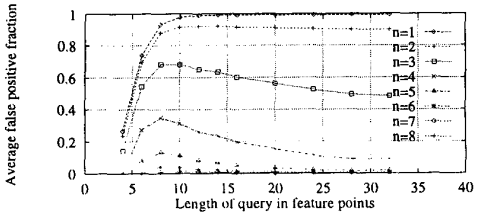
The results of our experiments show that n-gram indexing is a viable option for building music retrieval systems. Indexing using n-gram has the advantage of being efficient during both search and index construction processes. Sophisticated music analysis is not required, and combinatorial explosion problem is found to be relatively small. Our simulated queries on n-gram indices of six musical features show that the choice of n depends on the musical feature, but n is often smaller than 8.

We also found that if no musical feature "stopwords" are identified and removed, nonuniformly quantized Interval Sequences such as Profile are more discriminating than uniformly quantized ones such as CI5 and CI7. Hence, to design new features based on quantization of Interval Sequence, nonuniformly quantization to more than three levels can be a good choice.

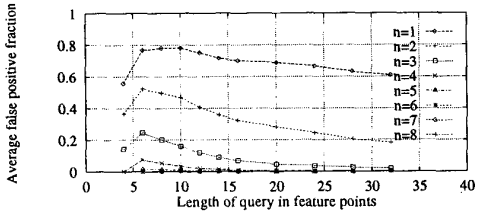
Our study on the relative dispersion on the fraction of



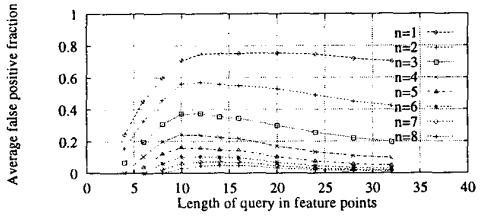
(a) Interval Sequence



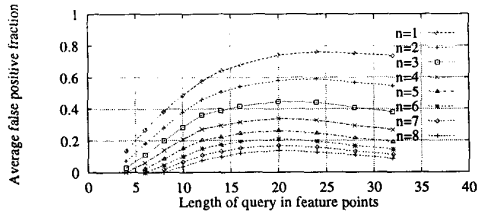
(b) Profile



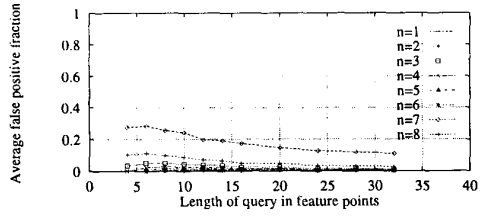
(c) CI3



(d) CI5



(e) CI7



(f) Note Duration Ratio Sequence

Figure 3: Fraction of false positives in the database under different query lengths

the database that matches the samples shows that Interval Sequence and CI3 are good in the sense that they "partition" the database more or less evenly at a reasonable query lengths. The feature Profile also shows a similar property although a longer query lengths are needed. Larger relative dispersions are found for Coarse Interval features and Note Duration Ratio Sequences; some of them match very few pieces in the database, while some match a lot. Musical "stopword" identification and removal are needed to make them more discriminating.

Besides reporting results on approximate n-gram matching of music features as described in Section 4, The authors are currently working on a number of issues in music retrieval. Some of them include the design of features that are suitable for handling polyphonic music, the study on the effect of combining different features in content-based music search, and the effect of different musical genres on music retrieval systems. Moreover, the design of indexing schemes besides n-gram are also being carried out.

7. ACKNOWLEDGMENTS

Special acknowledgments are given to CHENG Man Yee, HO Wai Shing, TANG Fung Michael, and YUEN Kin Wai for their help on the extraction of melodic lines from the music collection which made the experiments possible.

8. REFERENCES

[1] W. Jay Dowling. Scale and contour: Two components of a theory of memory for melodies. *Psychological Review*, 85(4):341–354, July 1978.

[2] J. Stephen Downie. Informetrics and music information retrieval: An informetric examination of a folksong database. In *Proc. 26th Annual Conf. of the Canadian Assoc. for Info. Sc. (CAIS 1998)*, 1998.

[3] Joo Ho Lee and Jeong Soo Ahn. Using n-grams for Korean text retrieval. In *Proc. SIGIR '96*, pages 216–224, 1996.

[4] Eleanor Selfridge-Field, editor. *Beyond MIDI: The Handbook of Musical Codes*. The MIT Press, 1997.

[5] Ching Y. Suen. *Computational studies of the most frequent Chinese words*. World Scientific, 1986.

[6] Esko Ukkonen. Constructing suffix trees on-line in linear time. In J. van Leeuwen, editor, *Algorithms, Software, Architecture: Information Processing 92*, volume 1, pages 484–492. Elsevier Science B.V., 1992.

[7] Chi Lap Yip and Ben Kao. A study on musical features for melody databases. In *Proc. DEXA 1999*, number 1677 in LNCS, pages 724–733. Springer-Verlag, 1999.