Proceedings of the International MultiConference of Engineers and Computer Scientists 2013 Vol I,
IMECS 2013, March 13 - 15, 2013, Hong Kong

Recommendation Method based on Rough Set for Sequential Data

Daisuke Nagakubo, Yasuo Kudo and Yoshifumi Okada

***Abstract*— Recently, recommendation technology, aimed at providing information suited to user preference, has become a very active research topic. Previously, we developed a recommendation system using rough sets taking sequential data as their object. We have attempted to apply this system to music recommendations based on chord progression patterns. However, because this method used fixed length subsequences as attributes for the rough sets, the musical meaning of the subsequences was completely ignored. In our current research, we extend this method to allow handling of variable length subsequences to generate musically meaningful subsequences. This paper evaluates the effectiveness of the method through music melody recommendations based on chord progression patterns. The relevance of the extracted patterns to musical meaning is examined using Japanese melodies. We show that this method improves upon the recommendation accuracy of a traditional method and that it extracts meaningful chord progressions that reflect the lyrics of a song.**

***Index Terms*— rough set, recommendation system, sequential data**

I. Introduction

Currently, recommendation technology is being actively researched to provide products and information (henceforth, the "object") matched to user interests and tastes for online shopping [1]. We have previously extended the information recommendation method using rough sets described by Kudo *et al*. [2] to take sequential data as its object and have reported the effectiveness of the extended method [3]. However, this method used fixed length subsequences as attributes for the rough sets and failed to consider consistency of meaning.

In our current research, we have extended our previous method to allow handling of variable length subsequence patterns having consistency of meaning, and we have built an information recommendation system to implement this method. This paper evaluates the usefulness of the suggestion system through music melody recommendations based on chord progression patterns.

Manuscript received December 26, 2012. This work was partly supported by Grant-in-Aid for Young Scientists (B) No.24700204 from MEXT Japan.

Daisuke Nagakubo is with College of Information and Systems, Muroran Institute of Technology, 27-1, Mizumoto-cho, Muroran 050-8585, Japan (e-mail: nagakubo@cbrl.csse.muroran-it.ac.jp).

Yasuo Kudo is with College of Information and Systems, Muroran Institute of Technology, 27-1, Mizumoto-cho, Muroran 050-8585, Japan (e-mail: kudo@csse.muroran-it.ac.jp).

Yoshifumi Okada is with College of Information and Systems, Muroran Institute of Technology, 27-1, Mizumoto-cho, Muroran 050-8585, Japan (corresponding author to provide phone: +81-143-5408; fax: +81-143-5408; e-mail: okada@csse.muroran-it.ac.jp)

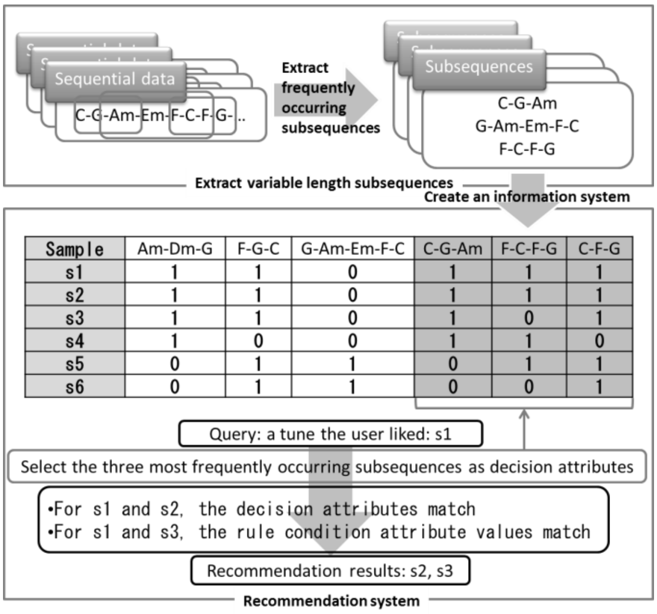

Fig. 1. The flow of information for recommendation in the current method

II. Rough Sets

Data concerning an object is often assigned as multiple attributes and their values. A table showing multiple attribute values of an object is called an information system. Our information system is defined by the four-tuple (U, AT, V, ρ). In this system, U represents the whole set of objects appearing in the system, AT represents the set of attributes, V represents the set of attribute values, and $\rho : U * AT \to V$ is a function that assigns attribute values to the attributes [4].

The attributes in this information system are divided between decision attributes, which divide the whole set of objects, and the other attributes, which are condition attributes. Here the information system is called a decision table. The collections of objects divided by decision attribute values are called decision classes. The smallest collection of condition attributes, which are required to determine decision class an object belongs to, is called the relative reduct. The rules concerning inclusion of attributes in the relative reduct and the values of those attributes used to determine decision class an object belongs to are called decision rules.

ISBN: 978-988-19251-8-3
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

III. The method

Figure 1 shows the flow of the melody recommendations. The method can be broadly divided into two parts: 1) construction of the information system using variable length subsequences and 2) melody recommendations using rough sets.

A. *Construction of the information system using variable length subsequences*

Because the sequential data has neither attributes nor attribute values, we cannot directly create the information system. Kaneiwa *et al*. [5] have solved this problem by extracting subsequences, which are fixed length patterns.

However, these fixed length subsequences are simply extracted by automatically shifting the window. Therefore, absolutely no consideration was given to the meaning of these sequences. Our method changes the extraction method from fixed length to variable length. We use the variable *n*-gram method to extract variable length subsequences [6].

The variable *n*-gram method is used to search similar character strings. This method starts by setting the subsequence length to one and then counts occurrences. When the number of occurrences reaches or exceeds a frequency threshold, the subsequence length is increased by one and the count of occurrences is repeated. By frequency threshold, we mean the lowest number of occurrences.

The above process is repeated until a subsequence with length n and a frequency lower than the threshold frequency occurs, and subsequences of length up to $n-1$ have been extracted. In this manner, with melodies as the object, each subsequence as an attribute, and the existence or non-existence of the subsequence in the melody as the attribute value, an information system is created, as illustrated in Figure 1.

B. *Melody recommendations using rough sets*

Melody recommendation is performed by the following steps:

1) The user selects one melody he/she likes. We refer to this melody as the query.
2) From the chord progression contained in the query, we select the three most frequently occurring chord progressions in order as decision attributes. The remaining chord patterns are set as condition attributes.
3) Melodies with matching decision attributes are extracted from the information system.
4) The relative reduct is found and decision rules are derived from the decision attribute and the condition attributes in Step 2.
5) The decision rule that has the highest value with regard to coverage average value is specified as the relative reduct.
6) A melody that matches the relative reduct decision rule derived in Step 5 and the query decision rule attribute values to a degree greater than a standard value are extracted. That standard value starts at 1.0. If a melody cannot be extracted at that level of matching, the standard level is iteratively reduced by 0.1 until matches are found (the lower limit is 0.7).
7) The melodies extracted in Steps 3 and 6 are output as recommended melodies.

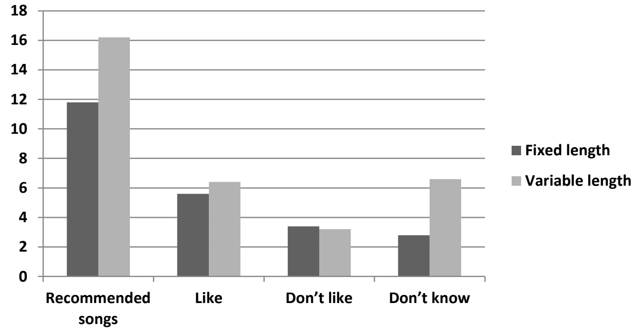

Fig. 2. Evaluation of the recommendation system

IV. Experiments

Two experiments were performed to evaluate the effectiveness of the current method. The first evaluates the recommendation results, and the second verifies whether the variable length subsequences are musically meaningful. Here we set the fixed length of the subsequence to three for the previous method.

A. *Evaluation of recommendation results*

Ten songs from ten different artists were randomly chosen from a web site [8]; yielding a total of 100 chord progressions. In the evaluation experiment, five men in their 20s used the proposed system. The recommendation results were subjectively evaluated. The evaluation for each recommended melody was performed by selecting one of three possible responses: “This is a song I like,” “This is a song I don’t like,” or “I don’t know this song.”

The results are shown in Figure 2. When variable length subsequences were used, liked melodies and unknown melodies both increased over the results for the case using fixed length subsequences. It is likely this resulted from the fact that not only more melodies were recommended but more frequent chord patterns, i.e., meaningful subsequences were used to make the recommendations.

B. *Evaluation of subsequences*

In this experiment, we used the lyrics and scores of five children’s songs and popular songs that were offered by a web site [9]. Here we evaluated whether musically meaningful chord patterns would actually be extracted in the cases of variable length subsequences for the current method and a fixed length subsequence for the previously used method. The correspondence between the extracted subsequences and the coherence of the song’s lyrics and the musical notes was examined.

We call a chord pattern taken from the unity of the words and the music a comparison subsequence. The comparison subsequence for the words was generated as follows. First, when considering lyrics, wherever there was a vocal pause and the immediate previous word was a verb, adverb, or noun we considered the section up to that position as one block. We extracted the chord pattern included in each block as a comparison subsequence.

We call this subsequence a lyric-partition subsequence. Similarly, with regard to the musical notes, a comparison sequence was generated in the following manner. First, chord patterns were identified by dividing the music at every second

ISBN: 978-988-19251-8-3
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

note. Next, each of these chord patterns was extracted as a comparison subsequence. We call this subsequence a note-partition subsequence.

Next, we compared both the fixed length subsequence extracted by the previous method and variable length subsequences extracted by the current method to the comparison subsequence. From the chord patterns for each song, we extracted fixed length sequences, variable length sequences, and comparison sequences and used precision and recall measures, as defined below, to perform an evaluation.

$$\mathrm{Pr}\,ecision = \frac{Cor}{Sys}$$

$$\mathrm{Re}\,call = \frac{Cor}{Std}$$

Here *Cor* represents the number of exact matches between the chord pattern of the extracted fixed or variable length subsequence and the extracted comparison subsequence *Sys* represents the number of extracted fixed or variable length subsequences, and *Std* represents the number of extracted comparison subsequences.

Figure 3 shows the results of comparing the chord patterns of the songs with the subsequences extracted by the fixed and variable length methods. When compared to lyric-partitioned subsequences, the score for variable length subsequences significantly surpasses the score for fixed length subsequences. This suggests that the new method appropriately extracts frequently occurring chord sequences. Furthermore, these chord patterns are very strongly related to meaningful units of the songs.

On the other hand, when compared to note-partitioned subsequences, fixed length performs slightly better than variable length. This is because the length of the note subsequences is set to two, which makes it very easy for these to occur in fixed length subsequences with a length of three.

V. Conclusion

Our current research extended our previous rough set information recommendation method by implementing the ability to handle variable length subsequence patterns. We learned through evaluation experiments that use of variable lengths demonstrates a higher accuracy level for recommendations and that variable length subsequences reflect the lyrics of songs in a meaningful manner. In future research, we will look into increasing the accuracy of evaluation for songs that are not known to the user.

Acknowledgment

This work was partly supported by Grant-in-Aid for Young Scientists (B) No.24700204 from MEXT Japan.

References

[1] Xiaoyuan Su and Taghi M. Khoshgoftaar: A Survey of Collaborative Filtering Techniques, Advances in Artificial Intelligence, Vol.2009, 2009.

[2] Yasuo Kudo, Shohei Amano, Takahiro Seino and Tetsuya Murai: A Simple Recommendation System Based on Rough Set Theory, Kansei Engineering International, Vol.6, No.3, pp.19-24, 2006.

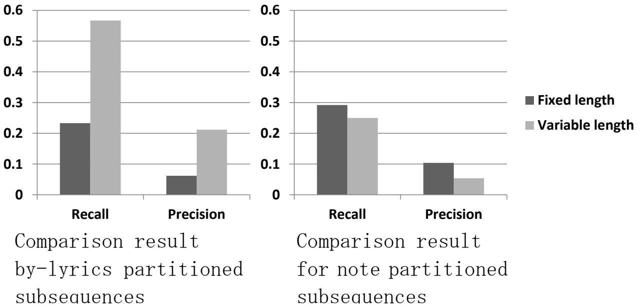

Fig. 3. Subsequence evaluation results

[3] D. Nagakubo, Y. Kudo and Y. Okada: A Music Recommendation System using Rough Set, In Proceedings of JSKE14, 2012 (in Japanese).

[4] Z. Pawlak : Rough Sets, Internat. J. Inform. Comput. Sci, Vol.11, No.5, pp.341-356, 1982.

[5] Ken Kaneiwa and Yasuo kudo: A sequential pattern mining algorithm using rough set theory, International Journal of Approximate Reasoning, Vol.52, Issue 6, pp.881-893, 2011.

[6] Siu, M. and Ostendorf, M.: Variable n-grams and extensions for conversational speech language modeling, IEEE Trans. on Speech andAudio Processing, Vol.8, pp.63–75, 2000.

[7] Yasuo Kudo and Takahiro Murai: An Evaluation Method of Relative Reducts Based on Roughness of Partitions, Int. J. Cognitive Informatics and Natural Intelligence, Vol.4, No.2, pp.50-62, 2010

[8] J-Total Music: http://music.j-total.net/ (in Japanese)

[9] Ototama: http://ototama.com/index.php (in Japanese)

ISBN: 978-988-19251-8-3
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)