A Melody Composer for both Tonal and Non-Tonal Languages

Coleman Yu, Raymond Chi-Wing Wong
Department of Computer Science and Engineering
The Hong Kong University of Science and Technology
cyuab@cse.ust.hk, raywong@cse.ust.hk

ABSTRACT

Song consisting of a melody part and a lyric part is very important to human. Song is more informative than a instrumental composition. People want to have their own songs for their special days. However, composing a song is difficult to people without music background because it requires some music knowledge and some composition writing skills. There is an algorithmic melody composer called "T-Music", which finds (or mines) the correlations between the melodies and the lyrics and uses these correlations to compose a melody for the input lyrics. These correlations are represented in ***f****requent* ***p****atterns (****fp****s). This paper presents the ways of enhancing T-Music. The original T-Music can only mine fps from songs. We propose two new methods to mine fps from instrumental compositions. We also introduce an optimal way of using fps mined from songs in one language to compose a melody for the input lyrics in another language.*

1. INTRODUCTION

There are a lot of people in the world without any music background who would like to compose songs for their important people and days. We observed that people could use their languages for communication. We would like to have an algorithm which takes lyrics as input and returns a good melody in order to help people without any music background to compose songs.

English speakers can input lyrics (in English) "London Bridge is falling down" to T-Music, and T-Music will return a melody as shown in Figure 1. Mandarin speakers can input lyrics (in Chinese) "当我还是一个懵懂的女孩" to T-Music, and T-Music will return a melody as shown in Figure 2. The melody returned by T-Music can be sung together with the input lyrics. T-Music is by no means to replace human composers. It is designed as a tool for assisting human composers and fostering creativity by providing a new method for composing melody.

1.1 Related Works

The study of using computers to compose melodies is called artificial music composition or algorithmic composition. There are many approaches to tackle this problem such as symbolic rule-based systems and evolutionary algorithms.

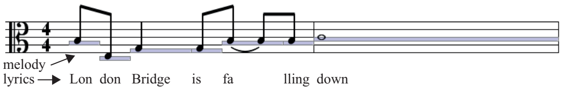


Figure 1. A melody returned by T-Music for English lyrics

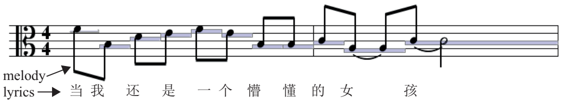


Figure 2. a melody returned by T-Music for Mandarin lyrics

To the best of our knowledge, [1] is the first study that includes the lyric-note correlations in the composition. The lyric-note correlation refers to the correlation between "notes of the melody" and "tones of lyrics". We are interested in such correlations that occur frequently. They are called **f**requent **p**atterns (fps).

An fp can be captured by absolute pitches and absolute tones [1]. An fp can also be captured by a pitch trend and a tone trend [2] . Trend refers to the pairwise differences of the absolute sequence. The "trend" representation is better than the "absolute" representation because same melodies which start at different pitches may sound similar to us. The first attempt to compose a melody for the input lyrics in one language based on fps that are mined from songs in another language was proposed in [3].

1.2 System Architecture

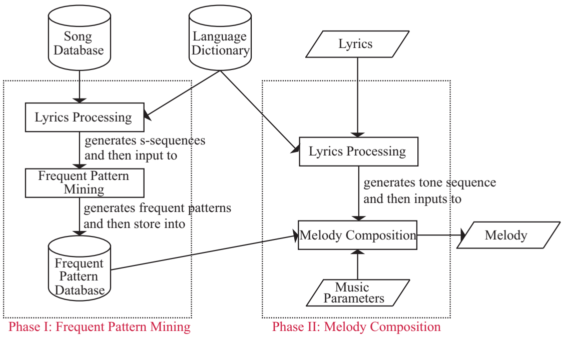

Song Database
Language Dictionary
Lyrics
Lyrics Processing
generates s-sequences and then input to
Frequent Pattern Mining
generates frequent patterns and then store into
Frequent Pattern Database
Lyrics Processing
generates tone sequence and then inputs to
Melody Composition
Melody
Music Parameters
Phase I: Frequent Pattern Mining
Phase II: Melody Composition

Figure 3. System Architecture of T-Music

Figure 3 shows the system architecture of T-Music. The system consists of two phases. They are "Frequent Pattern Mining" and "Melody Composition". "Frequent Pattern Mining" mines the fps from the song database and stores them in the FP database. For each song, it contains lyrics and a melody. By reading the language dictionary, the tone sequence of the lyrics can be obtained. A melody consists of notes, and a note consists of a pitch and a duration. Hence, a melody can be treated as consisting of a pitch sequence and a duration sequence. A song can be represented by these three sequences. These three sequences are together called as a s-sequence. Fps are mined from s-sequences. “Melody Composition” composes a melody for the tone sequence of the input lyrics based on the fps in the FP database. Some music rules such as harmonic rules are included in the composition process to ensure that the generated melody obeys the music rules and hence it is pleasant to human ears.

The main contributions of this paper are as follows. The original T-Music can only mine fps from songs in which lyrics must be present. However, instrumental compositions are more abundant than songs. Besides, the song files that are collected from the internet do not always have lyrics embedded. Finding corresponding lyrics and embedding the lyrics takes a long time. It is a shortcoming if we can only mine fps from songs. We propose two methods that can find fps from instrumental compositions in which lyrics are absent. In order to use the fps mined from songs with lyrics in language L_1 for the input lyrics in language L_2, a tone mapping from L_2 to L_1 is required. Random mapping of the tones in L_2 to that in L_1 can meet this purpose. However, the random mapped tones for the input lyrics may not use the mined fps well. It motivates us to design a method that uses the FP database as reference and decides the mapping of the tones in L_2 to that in L_1 such that the mapped tones for the input lyrics can use the fps well.

A video of composing a melody based on Mandarin lyrics, showing the major ideas in this paper, could be found at `https://vimeo.com/209610916`.

2. BACKGROUND

Spoken Language is known as speech. Speech is a sequence of sounds that conveys a meaning. A *syllable* is a basic unit in a speech. For example, in English, the phonetic of the word “music” contains two syllables.[1] They are [mjuː] and [zɪk]. In Mandarin, the phonetic of the word “音” contains 1 syllable. It is [yīn]. There is a mark on top of the i in [yīn]. It is a tone symbol. It tells us how to pronounce this syllable with the correct tone.

The languages can be classified as *tonal* or *non-tonal languages*. For both kinds of language, there are tone variations. Even in non-tonal languages, people do not speak in monotone. However, if someone speaks a sentence in monotone in English, others can still understand. The meaning of the sentence will not be ambiguous. In tonal languages, the tone of a word is important for distinguishing the word from another.

2.1 Tonal Languages

In English, a word “men” with phonetic [men] is different from another word “man” with phonetic [mæn] because they have different vowels [e] and [æ]. Using different tones to pronounce them will not change their meanings.

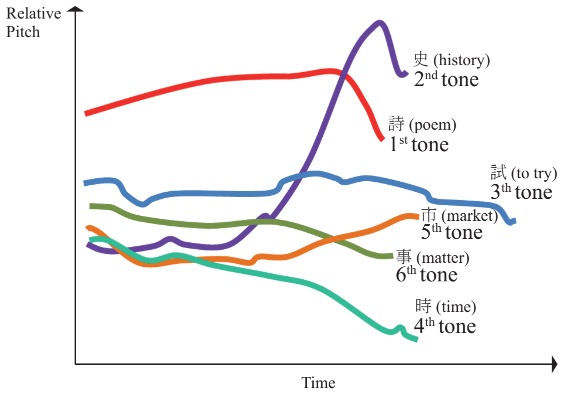
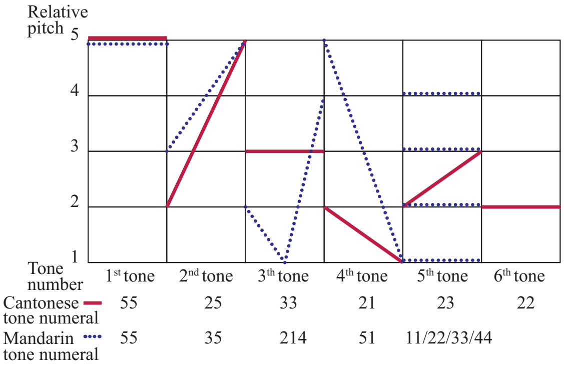


Figure 4. Cantonese Tone Contours



Tone number	1st tone	2nd tone	3th tone	4th tone	5th tone	6th tone
Cantonese ▬ tone numeral	55	25	33	21	23	22
Mandarin ···· tone numeral	55	35	214	51	11/22/33/44	

Figure 5. The six Cantonese tones and the five Mandarin tones

In Cantonese, which is a tonal language, a word conveys different meanings if it is pronounced in different tones. Cantonese has six tones [4]. Figure 4 shows the six tones and meanings of a word [si] if it is pronounced in these six tones. It is modified from [5].

A tone can also be represented in tone numeral. By setting the highest pitch as 5 and the lowest pitch as 1, and dividing the pitch spectrum uniformly, a tone can be represented by a tone numeral where the first digit represents how high the tone starts and the last digit represents how high the tone ends. For example, the 1st tone in Cantonese starts and ends at almost the highest pitch. The tone numeral of the 1st tone is 55. Middle digit can also be included if the tone changes non-linearly in the middle. The tone numerals of the remaining five tones are 25, 33, 21, 23 and 22 [6]. It is shown in Figure 5.

Mandarin has four formal tones. They are 55, 35, 214 and 51 [7]. There is a special tone called *neutral tone* in Mandarin.[2] When a syllable is associated with it, the syllable is pronounced shorter and unstressed. It is not considered as a formal tone because its pitch value is determined by the tone of the preceding syllable [8]. The five tones (including the natural tone) are shown in Figure 5.

2.2 Non-Tonal Languages

In English, which is a non-tonal language, has three kinds of stress, namely the primary stress, the secondary stress and the non-stress. Consider a word “education”, which

[1] Chinese phonetic is also called pinyin.

[2] In Chinese, it is called “qīng shēng (輕聲)”, which literally means “light tone”.

have four syllables [e-jə-kɑ-ʃən], we stress on the [kɑ], stress lightly on [e] and non-stress on [jə] and [ʃən]. These three stresses can be treated as three tones.

Japanese, which is also a non-tonal language, has two kinds of stress. They can be treated as two tones.

2.3 Summary

Given a word, we can obtain its (syllable, tone) sequence by reading a dictionary. We are only interested in the tone part but not the syllable part. For the input lyrics, we can obtain a sequence of consecutive words by applying word segmentation.[3] We can obtain a tone sequence for the input lyrics.

3. PHASE I: FREQUENT PATTERN MINING

We will discuss how to represent and mine the fps from songs. This task is a modification of the problem of mining sequential patterns, which was originally proposed by [9].

3.1 Mining Frequent Patterns from songs

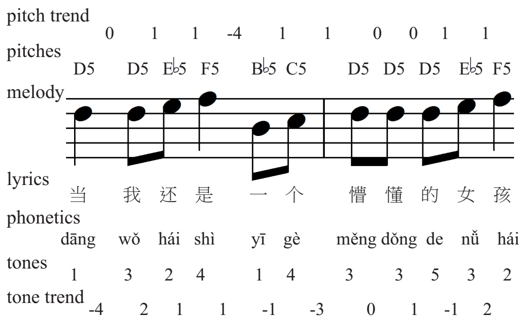


Figure 6. A melody segment from a Mandarin song

Figure 6 shows a segment of a melody which comes from a Mandarin song called "huī zhe chì bǎng de nǚ hái (挥着翅膀的女孩)". A melody can be converted to a pitch sequence and a duration sequence. T-Music uses both "the fps between the tone part and the pitch part" AND "the fps between the tone part and the duration part" to compose a melody. Since the technique of mining and using the former fps is similar to that for the later one, we omit the manipulation of the duration part.

Its pitches < D5, D5, ..., F5> are represented in letter names. By analysing the note distribution, we know that this song is in Bb Major. Hence, it is < mi5, mi5, fa5, so5, do5, re5, mi5, mi5, mi5 ,fa5 ,so5 > in the sol-fa name representation[4]. The number next to the sol-fa name of a note represents which octave the note is in.[5] The trend of this sequence is <0, **1**, ..., 1>. The 1st "1" indicates that the 3rd note is 1 sol-fa name higher than the 4th note.

A tone number sequence can be obtained from the lyrics. According to Figure 5, the 1st tone is the highest tone, the 2nd tone is the second highest tone and so on.[6] The relative tones are <5, 1, 3, 4, 5, 4, 1, 1, 2, 1, 3>. Its tone trend can be computed and is shown in Figure 6.

Given a song database D, we want to find out the fps of tone trends and pitch trends. A *p-pattern* p consists of a tone trend tt and a pitch trend pt, $p = (tt, pt)$. For example, (<1, 1>, <1, -4>) occurs one time and (<2>, <1>) occurs two times in the melody segment in Figure 6. If p occurs at least a threshold called "specific frequent threshold" in a song s, p is deemed to be specific frequent w.r.t. s. If p is specific frequent in not less than a threshold called "overall frequent threshold" songs in D, p is deemed to be overall frequent or frequent in short (w.r.t. D). Our goal is to mine the frequent p-patterns (fp in short) from D. This task can be done by a frequent sequence mining algorithm [10]. The mining algorithm works efficiently on this kind of music data. This is because there are always some notes in the songs that are not associated with tones. The patterns that consist of non-continuous tone trend would not be further processed. This method can be used for any languages, not just Mandarin.

3.2 Mining Frequent Patterns from instrumental compositions

Most of the music on the internet do not have lyrics embedded. It encourages us to develop two methods to mine fps from instrumental composition.

3.2.1 *Method emphasizing the original fps*

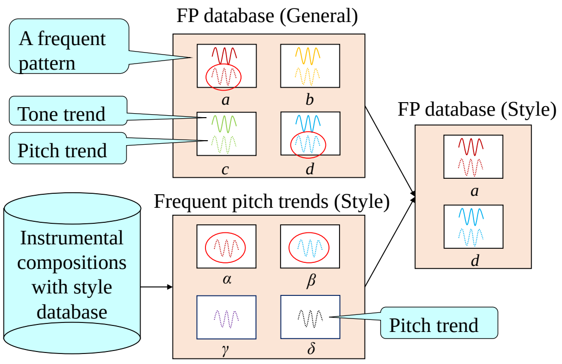


Figure 7. Extract Existing Frequent Patterns Consistent with Frequent Pitch Trends mined from Instrumental Compositions

This method is shown in Figure 7. Fps are mined from songs and are stored in "FP database (General)". Since instrumental compositions does not contain lyrics, the original mining algorithm cannot mine anything from them. However, the frequent pitch trends can still be mined from them and they are stored in "Frequent pitch trends (Style)". "Frequent pitch trends (Style)" is used as a selector and it selects those fps in "FP database (General)" that have pitch trend in the "Frequent pitch trends (style)". The selected fps are stored in "FP database (Style)" For example, a has a pitch trend that is the same as α. Hence, a is selected and is stored in "FP database (Style)". b has a pitch trend that

[3] Word segmentation is a trivial task for English because the words in a English sentence are separated by spaces. For other languages, it can be a difficult task. For example, Thai, as there is no inter-word spacing.

[4] "so" can also be written as "sol".

[5] An octave starts at a C note and ends at a B note.

[6] For simplicity, we perceive the neutral tone to have the second lowest pitch among the five tones. It is not the lowest because the 4th tone changes from the highest pitch to the lowest pitch which gives us a strong perception that the 4th tone is indeed the lowest tone.

is not the same as any pitch trend in “Frequent pitch trends (Style)”. Hence, it is not selected. This method emphasizes the original fps in the sense that “FP database (Style)” is a subset of “FP database (General)”.

3.2.2 Method emphasizing the newly mined frequent pitch trends

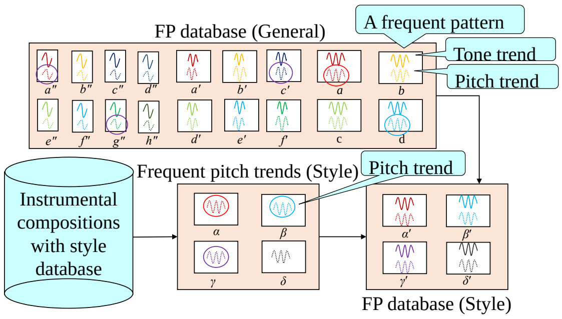


Figure 8. Tone Filling for the new mined frequent pitch trends

This method is shown in Figure 8. The “FP database (General)” shown in Figure 8 is identical to that shown in Figure 7. The former one also shows the shorter fps. According to the Apriori property [9], any subset of frequent itemset must be frequent. It implies that a frequent pattern p_1 consists of (1) a tone trend which is a subsequence of a tone trend of a frequent pattern p_2 and (2) a pitch trend which is also a subsequence of a pitch trend of p_2. p_1 is called a sub-pattern of p_2. $a', b', ..., f'$ are the sub-patterns of $a, b, ..., d$. $a'', b'', ..., h''$ are the shorter sub-patterns of $a, b, ..., d$. The frequent pitch trends are stored in “Frequent pitch trends (Style)”. For each newly mined frequent pitch trend, we guess its corresponding tone trend based on “FP database (Style)” to create an fp. For example, we guess the corresponding tone trend of α as follows. There is an fp, says a, that has a pitch trend equal to α. Hence, we guess the corresponding tone trend of α is the tone trend of a. For γ, there is no fp that has the same pitch trend as γ. Hence, we try to construct γ by concatenating the pitch trends of the shorter fps. We find that γ can be formed by concatenating the pitch trends of a'', c', and g''. We guess that the corresponding tone trend of γ is a sequence concatenated from the tone trends of a'', c', and g''. This method emphasizes the newly mined frequent pitch trends in the sense that “FP database (Style)” has the same size of “Frequent pitch trends (Style)”.

4. PHASE II: MELODY COMPOSITION

In Figure 3, “Melody Composition” reads “Frequent Pattern Database” to compose a melody for the input lyrics and music parameters which specify how the generated melody sounds like such as key signature and tempo.

4.1 Composing a melody using fps in the same language as the input lyrics

For an input lyrics, we can obtain its tone sequence and hence tone trend tt_l. Our goal is to construct the corresponding pitch trend pt_l by referencing the FP Database and use pt_l to compose a melody. pt_l indicates that how the pitches of the melody should be changed.

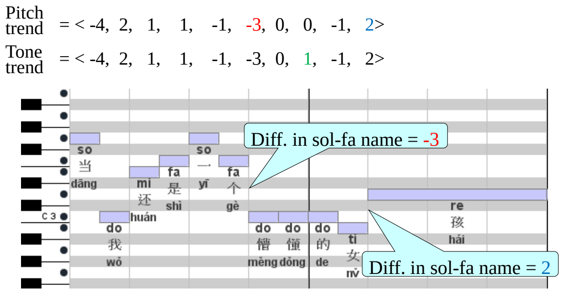


Figure 9. Compose a melody for the input lyrics

pt_l can be constructed as follows. The FP Database can be treated as a multi-map M. A map is a data structure that allows us to retrieve a value by a key quickly. A multi-map is a special map that the keys are not unique. An fp consists of a tone trend tt, a pitch trend pt and a support τ (i.e., frequency). For each frequent pattern stored in M, tt is used as a key and (pt, τ) is used as a value. The keys are not unique since fps can have the same tone trend but different pitch trends. We use tt_l as the key to retrieve the corresponding (pt, τ) tuples in M. These tuples are sorted in descending order of τ. A tuple in the top-k tuples is selected uniformly and randomly and its pitch trend is set to be pt_l. k is set by a user. By doing this, a pattern with a very large support does not dominate other patterns with a relatively small support.

If no tuple can be retrieved for tt_l, we will employ the divide and conquer idea. We divide the tt_l into at most three shorter tone trends, apply the same procedure on them and return the concatenation of the results of the sub-problems. If the input tone trend with a length of 1 (i.e., the base case) does not have the corresponding pitch trend in M , we simply set the pitch trend to be the input tone trend. There is no concatenation requirement for two adjacent pitch trends. This is because a pitch trend is not an absolute sequence.

Figure 9 shows the composition for the input Mandarin lyrics. The tones of the input lyrics is <1, 3, ..., 2>. The relative tones is <5, 1, ..., 3>. tt_l is stated in the figure. We can then find pt_l. We observed that tt_l and pt_l of lyrics are similar (with only one entry different). It matches our intuition that the sound sequence of the melody usually matches that of lyrics so that lyrics can be sung together with the melody.

We can generate the melody according to pt_l from the ending note. The ending note of a sentence is determined by the first note of its next sentence. The last sentence in the melody is set as the tonal note of the key signature. In Figure 9, the generated melody is in C Major. The ending note in the sentence in the figure is D3, which is a re note in the scale. In a major scale, the separations of adjacent notes in half step is ss^M = <2, 2, 1, 2, 2, 2, 1>. For example, there are 2 half steps between the 1st note (do note) and the 2nd note (re note). The melody is generated note by note from the ending note. For example, the 10th entry in the pitch trend (i.e., 2) indicates that the previous note should be 2 sol-fa name lower than the current re note. According to ss^M, this note is 3 half steps lower than the current note. [7] It is a ti note in the scale.

[7] 3 = 2 + 1. There are 2 half steps between the re note and the do note AND 1 half step between the do note and the lower ti note.

In a minor scale, the separations of adjacent notes in half step ss^m = <2, 1, 2, 2, 1, 2, 2> is used instead. ss^m is the same as ss^M but with different starting positions.

Besides, the generation of a pitch trend for a sentence in the lyrics does not depend on the FP database only. It also depends on the pitch trends of other sentences so that the melody generated for one sentence is coherent with that of other sentences.

4.2 Composing a melody using fps in different language with the input lyrics

We may want to use the fps mined from songs in language L_2 to compose the melody for the input lyrics in another language L_1. In order to do this, we need to map the tones in L_1 to that in L_2. For the tone sequence ts of an input lyrics in L_1, we can do the tone mapping from L_1 to L_2 on ts to generate a new tone sequence ts'. With ts', we can apply the same "Melody Composition" phase by using the FP database mined from songs in L_2.

A tone in L_1 maps to a tone (or tones) in L_2 that they have a similar tone counter. Besides, we should also map the tones in L_1 to that in L_2 uniformly.

4.2.1 Tone Mapping between Japanese and Mandarin

For example, Japanese has two tones, namely the low pitch tone (the l tone) and the high pitch tone (the h tone) while Mandarin has 5 tones. We denote the lowest Mandarin tone as 0, the second lowest tone as 1 and so on. The l tone can be mapped to the 0th and 1st. The h tone can be mapped to the 2th, 3st and 4nd tones. [8] The two Japanese tones can be mapped to the six Cantonese tones in a similar way.

4.2.2 Tone Mapping between Thai and Mandarin

There are five tones in Thai. They are <323, 21, 41, 35, 213> [11]. To map the tones in Thai to that in Mandarin, we can use the L-1 distance as the similarity measure. The L-1 distance between the 4th tone in Thai (i.e., 35) and the 2nd tone in Mandarin (i.e., 35) is the smallest, which is 0, among the L-1 distances of the possible 5 × 5 tone pairs. These two tones are mapped. The L-1 distance between the 5st tone in Thai (i.e., 213) and the 3rd tone in Mandarin (i.e., 214) is the lowest, which is 1, among the possible 4 × 4 tone pairs. They are again mapped. The tones in Thai can be mapped to that in Cantonese in a similar way. Since there are six tones in Cantonese, we need to map the last unmapped Cantonese tone to the most similar tone among the five Thai tones after we have built the five tone mappings between Thai and Cantonese.

4.2.3 Optimal Mapping

If the number of tones in L_1 is smaller than that in L_2, we need to choose which tones in L_2 that a tone in ts should map to. Suppose L_1 is Japanese and L_2 is Mandarin, we need to decide whether a particular l tone in ts should map to the 0th or 1st tone in Mandarin. Random mapping was proposed to solve this [3]. For the language which only has a few tones, it is easier for the composer to write the lyrics on the melody. It is because a spoken sound in the language with a few tones can suit to many different pitches while that in the language with many tones can only suit to a few pitches.

ts' that is constructed by random mapping may not use the FP database well. For example, a Japanese tone sequence ts^J = <h, l, h, h, l, h> can be mapped to either one of Mandarin tone sequences ts_1^M = <4, 1, 2, 3, 0, 4> and ts_2^M = <3, 1, 4, 4, 1, 3>.

Assume that the tone trend of ts_1^M does not appear in the FP database but that of ts_2^M appears in the FP database. If ts^J is mapped to ts_1^M, we can find an fp that its pitch trend can be used to compose a melody for ts^J. If ts^J is mapped to ts_2^M, we cannot find such fp. It motivates us to design a method called optimal mapping that the mapping is done by referencing the fp database. Optimal mapping allows us to find the pitch trend(s) in the same fp with the tone trend(s) of the Mandarin tone(s) which is (are) resulted by possible Mandarin tone mapping(s) of ts^J.

Lemma 1 can facilitate us to design the optimal mapping method. Figure 10 is an example of Lemma 1.

Lemma 1. *A Mandarin tone trend can be generated from at most 3 Japanese tone sequences, no matter how long the Mandarin tone trend is.* □

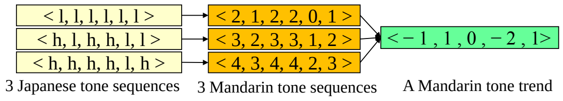


Figure 10. A Mandarin tone trend that can be generated from 3 Japanese tone sequences

The optimal mapping method is shown in Figure 11. We can generate at most 3 Japanese tone sequences for each Mandarin tone trend in "FP database (Mandarin)". Given the Japanese input lyrics, we can use its Japanese tone sequence to find its corresponding Mandarin tone trend and frequent pattern fp by following the links between "Japanese tone sequences" and "FP database (Mandarin)" and use the pitch trend of fp to compose a melody on input Japanese lyrics.

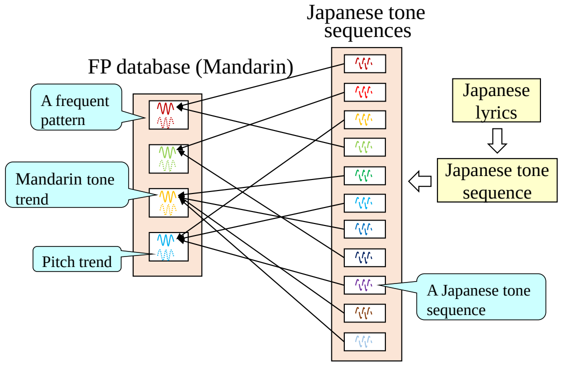


Figure 11. Optimal mapping from Japanese tones to Mandarin tones

Finally, we give the proof of Lemma 1. Our methodology of using the fps in L_2 to compose lyrics in L_1 can be modified and used for any pair of two languages.

Proof. *Given a Mandarin tone trend tt^M, its corresponding Mandarin tone sequences can be generated by starting from 0 to 4 and applying tt^M. Some of the entries of*

[8] The 3th lowest tone (214) is mapped to the h tone because it is sharply increasing in the later part of the tone.

the resulted sequences may underflow or overflow the legal range of 0..4. Those resulted sequences with all its entries fall in the legal range are the Mandarin tone sequences of tt^M. There are at most 5 Mandarin tone sequences for tt^M. A Mandarin tone sequence can be converted to a Japanese tone sequence. The resulting Japanese tone sequences may not be all distinct. Hence, tt^M can be generated by at most 5 Japanese tone sequences. We can have a tighter upper bound.

The Mandarin tone sequences of tt^M are translated up or down of each other. Let d be the difference of the largest entry max and the smallest entry min of any one of the sequences (i.e., d = max - min). There are 5 - d vertical translations for this sequence to form all the Mandarin tone sequences of tt^M that each of them does not overflow or underflow. For example, in Figure 10, a Mandarin tone sequence <4, 3, 4, 4, 2, 3>, $d = 4 - 2 = 2$, has 3 possible transactions that do not occur overflow or underflow, namely +0, -1 and -2.

In order to generate tt^M from 5 Japanese tone sequences. d must be 0. $d = 0$ implies that all the entries in any one of the Mandarin tone sequences have the same value. Hence, the resulting Japanese tone sequences converted from these Mandarin tone sequences consist of all l or all h. There are indeed 2 but not 5 Japanese tone sequences.

In order to generate tt^M from 4 Japanese tone sequences. d must be 1. $d = 1$ implies that the entries have only two distinct values in any one of the Mandarin tone sequences. Hence, the resulting Japanese tone sequences converted from these Mandarin tone sequences consists of all l [9] *or all h* [10] *or a mix of l and h* [11]*. There are indeed 3 but not 4 Japanese tone sequences.*

In Figure 10, tt^M is generated from 3 Japanese tone sequences. □

Similar lemma as Lemma 1 can be constructed for Cantonese which could be found in Lemma 2 as follows. Figure 12 is an example of Lemma 2.

Lemma 2. *A Cantonese tone trend can be generated from at most 4 Japanese tone sequences, no matter how long the Cantonese tone trend is.* □

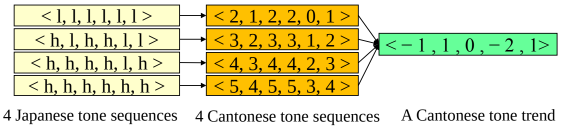


Figure 12. A Cantonese tone trend that can be generated from 4 Japanese tone sequences

5. CONCLUSION AND FUTURE DIRECTIONS

We have introduced a novel algorithmic composer called T-Music, which uses the correlations between melodies and lyrics to compose a melody for the input lyrics. Two new methods has been proposed to mine fps from instrumental compositions. An optimal method of composing a melody for input lyrics in one language based on the fps in another language has been presented.

There are a lot of possibilities to further improve T-Music. For example, we may also consider using the correlations between melodies and the syllables of lyrics to compose a melody. Besides, the number of notes in the generated melody is the same as the number of tones in lyrics. If T-Music can generate a longer melody by introducing some notes that does not associate with any tones, it will involve more varieties in the melody.

Acknowledgments

We are grateful to the anonymous reviewers for their constructive comments on this paper. The research is supported by HKRGC GRF 16219816.

6. REFERENCES

[1] C. Long, R. C. W. Wong, and R. K. W. Sze, "T-Music: A melody composer based on frequent pattern mining," in *2013 IEEE 29th ICDE*, pp. 1332–1335.

[2] ——, "Trend-MC: A Melody Composer by Constructing from Frequent Trend-Based Patterns," in *2015 IEEE ICDMW*, pp. 1628–1631.

[3] P. Pengcharoen, "A powerful tool of composing melody for different languages," Master's thesis, Electronic and Computer Engineering, HKUST, 2015.

[4] P. C. K. Chu and M. Taft, "ARE THERE SIX OR NINE TONES IN CANTONESE?" *PLRT 2011*, 2011.

[5] A. L. Francis, V. Ciocca, L. Ma, and K. Fenn, "Perceptual learning of Cantonese lexical tones by tone and non-tone language speakers," *Journal of Phonetics*, vol. 36, no. 2, pp. 268–294, 2008.

[6] C. Y. C. Flynn, "Intonation in Cantonese," Ph.D. dissertation, University of London (School of Oriental and African Studies), 2001.

[7] W.-S. Lee and E. Zee, "Standard Chinese (Beijing)," *Journal of the International Phonetic Association*, vol. 33, no. 01, pp. 109–112, 2003.

[8] J. Wang, "The neutral tone in trisyllabic sequences in Chinese dialects," in *International Symposium on Tonal Aspects of Languages: With Emphasis on Tone Languages*, 2004.

[9] R. Agrawal and R. Srikant, "Mining sequential patterns," in *Data Engineering, 1995. Proceedings of the Eleventh International Conference on.* IEEE, 1995, pp. 3–14.

[10] J. Ayres, J. Flannick, J. Gehrke, and T. Yiu, "Sequential pattern mining using a bitmap representation," in *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining.* ACM, 2002, pp. 429–435.

[11] P. Ladefoged and K. Johnson, *A course in phonetics*. Nelson Education, 2014.

[9] The two values both belong to the l group.

[10] The two values both belong to the h group.

[11] The larger number (It must be 2.) maps to h and the smaller number (It must be 1.) maps to l.