The extraction of beautiful sound patterns from Sunthorn Phu's Poem using machine learning technique and internal rhyme rule

Pattarakorn Suksanguan [a,1], Sajjaporn Waijanya [a,2,*], Nuttachot Promrit [a,3]

[a] Center of Excellence in AI and NLP, Department of Computing, Faculty of Science, Silpakorn University, Nakhon Pathom, Thailand
[1] suksanguan_p@silpakorn.edu; [2] waijanya_s@silpakorn.edu; [3] promrit_n@silpakorm.edu
* corresponding author

ARTICLE INFO

Article history
Received October 30, 2020
Revised December 20, 2020
Accepted January 21, 2021
Available online July 31, 2021

Keywords
Machine Learning
Apriori Algorithm
Rule-Based
Sunthorn Phu
Sound Pattern

ABSTRACT

The melodious poems have been written from the distinctive features of poetry or based on each country's typical style. Especially, Thai poems which composed by the use of specific forming, such as Internal Rhyme to develop melodiousness. The most attractive and well-known poems were composed by a genius Thai poet named Sunthorn Phu. He is a role model for Thai poets. UNESCO honored him as the world's great poet and the best role model in poetry works. In this article, we proposed extracting 15,796 sentences (Waks) of the beautiful sound patterns of Phra Aphai Mani's tales by machine learning technology in conjunction with the rules of internal Rhyme Klon-Suphap by using the Apriori Algorithm. The extraction of vowel rhymes separated by a group of Waks including 1) Poem Wak No. 1; 2) Poem Wak No. 2; 3) Poem Wak No. 3; and 4) Poem Wak No. 4. In this article, "Wak" means sentence. The created tool can extract the internal rhyme patterns and the 25 popular pattern vowels. The popular pattern illustrates the melodiousness of the Poem and sets up a standard of how to melodiously compose a poem. Then, the evaluation of the experiments was done by using 144 Waks selected from the extraction of the beautiful patterns and evaluated by the consistency score from 3 experts. The average accuracy score resulted in 95.30%.

1. Introduction

Klon-Suphap (Klon-Pad) is the most popular Poem in Thailand. It has simple and uncomplicated prosody all together with rhythm and sound patterns which create melodiousness. The structure of Klon-Suphap External Rhyme Prosody is shown in Fig. 1 [1]. External Rhyme occurs in relation between Wak and another, forward to each other, and Rhyme between chapters in other poems. It rules strict of prosody. A, b, c, and d is rhyme position, link by lines. The best of Klon-Suphap poems are the poems composed by Sunthorn Phu. The masterpiece of Sunthorn Phu is an epic poem named Phra Aphai Mani. Phra Aphai Mani is one of the most famous poems and has distinctive internal rhyme prosody. The internal rhyme prosody for Thai Poetry Klon-Suphap is shown in Fig. 2, for example, in Fig. 3 and Fig. 4, respectively. Fig. 3 and Fig. 4 show the phonetic alphabet of the Poem Phra Aphai Mani that has internal Rhyme, assonance, and alliteration. These components help develop the melodiousness of the Poem.

To compose a melodious poem, there must be some references based on Sunthorn Phu's prototype. This Sunthorn Phu's style poem has been recognized as one of the most beautiful and popular poems [2] which was composed correctly according to the prosody. However, having only the correct external

Rhyme of prosody still doesn't generate melodiousness. Sunthorn Phu performed the Poem more eargasmic by adding internal Rhyme to make the more dazzling Poem. In comparison, the ratio of a Wak with internal Rhyme is much higher than a Wak which doesn't have internal Rhyme in a poem. Therefore, the importance of internal Rhyme shouldn't be overlooked.

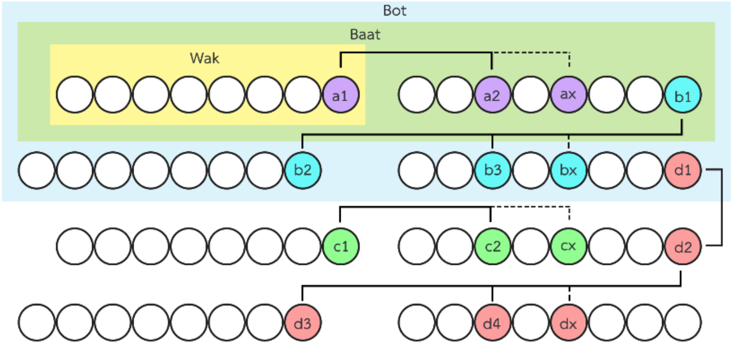


Fig. 1. External Rhyme Prosody of Klon-Suphap Poetry

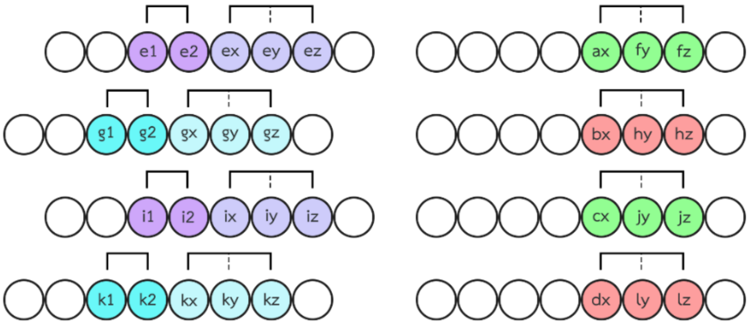


Fig. 2. Internal Rhyme Prosody of Klon-Suphap Poetry

He smiled to explain.	Don't run away, Grandpa, grandma Don't need to be sad.
เจ้าพราหมณ์ยิ้มพริ้มพรายธิบายบอก	ไม่หนีดอกยายตาอย่าโศกศัลย์
caw^3-phra;m^1-jim^4-phrim^4-phra;j^1-thi^4-ba;j^1-b@;k^2	maj^3-ni;^5-d@;k^2-ja;j^1-ta;^1-ja;^2-so;k^2-san^5
He fights prosecution.	No punishment, nothing.
ข้อคดีมีอยู่จะสู้กัน	อันโทษทัณฑ์สิ่งใดก็ไม่มี
kh@;^3-kha^4-di;^1-mi;^1-ju;^2-ca^2-su;^3-kan^1	?an^1-tho;d^3-than^1-sing^2-daj^1-k@^3-maj^3-mi;^1

Fig. 3. The example of assonance in Klon-Suphap Poem

He smiled to explain.	Don't run away, Grandpa, grandma Don't need to be sad.
เจ้าพราหมณ์ยิ้มพริ้มพรายธิบายบอก	ไม่หนีดอกยายตาอย่าโศกศัลย์
caw^3-phra;m^1-jim^4-phrim^4-phra;j^1-thi^4-ba;j^1-b@;k^2	maj^3-ni;^5-d@;k^2-ja;j^1-ta;^1-ja;^2-so;k^2-san^5
He fights prosecution.	No punishment, nothing.
ข้อคดีมีอยู่จะสู้กัน	อันโทษทัณฑ์สิ่งใดก็ไม่มี
kh@;^3-kha^4-di;^1-mi;^1-ju;^2-ca^2-su;^3-kan^1	?an^1-tho;d^3-than^1-sing^2-daj^1-k@^3-maj^3-mi;^1

Fig. 4. The example of alliteration in Klon-Suphap Poem

Internal Rhyme is not restricted in both assonance and alliteration. It is a rhyme within a Wak. There is a play on words with a particular rhyme sound than normal to produce melodies and beautiful sounds. Internal Rhyme comprises assonance and alliteration. In some positions where assonance cannot be used, alliteration might be better. The popular technique used in internal Rhyme is assonance due to the various word choices. On the other hand, it is more challenging to use alliteration due to the limitation of words. Thus, a poet who writes a poem that includes both assonance and alliteration can be called a gifted poet. In addition to composing poems with rhymes, there is an adaptation of rhyme usages combining with songwriting by composing a song that has rhymes to develop the song's melodiousness. For example, the lyrics from a song called "The run ground whale" (วาฬเกยตื้น -GUNGUN) in Thai is "เพราะฉันนั้นเป็นวาฬที่เกยตื้นน้ำ**ตาย** ใจ**สลาย**แหลกลงไปใต้ทะเล", the phonetic alphabet is "phr@^4 chan^5 nan^4 pen^1 wa;n^1 thi;^3 k#;j^1 tv;n^3 nam^4 **ta;j^1** - caj^1 sa^2 **la;j^5** lx;k^2 long^1 paj^1 taj^3 tha^4 le;^1". The song's translated lyrics are "Because I am the run ground whale to die - My heart was broken at the bottom sea". From the example, the underlined words are internal rhymes and the bold words are external rhymes.

The extraction of the melodious sound patterns of poems is a challenge in natural language processing (NLP) because computers require to be developed to understand the prosody structure of the Poem (Klon-Suphap) and the position of the pair of sounds that make it melodious; see the reference in Fig. 2.

The melodiousness of poems could be analyzed in many terms, such as rhythms and syllables used in the Poem, words, and meanings used in the Poem. Many foreign poems (non-Thai language poem) used Heuristics for analyzing rhythms and syllables in Brazilian poems [16] and use rule-based for analyzing syllables in Arabic poems [17]. In Chinese Poem, the Apriori algorithm [19] was used to analyze the links between characters and analyze the association between ancient poetry [20]. In addition, this algorithm was also used to analyze ancient Chinese words by detecting words and finding new words in the ancient Chinese library [21] in conjunction with the use of BiLSTM (Bidirectional Long Short Term Memory) [22] to detect neologism. In Thai Poem, melodiousness depends on prosodies and what makes poems even more melodious is the internal rhymes. Therefore, verses must conform with prosodies [23]. Backus Naur Form (BNF) [24] technique can be used to verify verse validity. This technique verifies the word count and the melodiousness of prosodies but still lacks verification of the melodiousness of internal rhymes.

In the past, there was a similar study of Sunthorn Phu's Poem [15] focusing on Klon Niras, which is similar to Klon-Suphap. That research's main objective was to study the genuine poetry style of Sunthorn Phu by using the manual counting method, which analyzes the various characteristics that appear in the verse of Niras poem but did not identify the melodiousness pattern. This research presents the extraction of beautiful sound patterns from Sunthorn Phu's Poem to reflect the melodious sound of the internal Rhyme that Sunthorn Phu used in his composing. Therefore, nobody has researched this subject before.

This research extracts the melodious sound patterns of Phra Aphai Mani tales that Sunthorn Phu frequently used to compose poems. This study aims to illustrate the melodiousness of the Poem and set up a standard of how to compose a poem melodiously. We found some researcher works for poem domain such as classify the poem [3]–[6], extract features poem [7], poetry generation [8]–[10], evaluation poem [11], translated poetry [12]–[14], poem entity recognition [15], and analysis of the melodiousness of the Poem [16]–[18]. Nobody has researched extracting the melodious sound patterns before. Our analysis of the internal rhyme poetry structure was done by using Apriori Algorithm. The accuracy of the results was evaluated by many experts, including members of the Thai Poetry Institute, the Poetry Society members, the Language and Books Association, and the Vice President of the Thai Contemporary Poets Association. The extraction of the melodious sound patterns from a masterpiece of the role model poet resulted in the model, which can be used to create the automatic tools. The tools can examine and recommend the method for composing the melodious Poem (Klon-Suphap).

2. Method

2.1. Process Overview

This research was divided into two main processes: 1) the preparation process and 2) the pattern recognition process. The process overview is shown in Fig. 5.

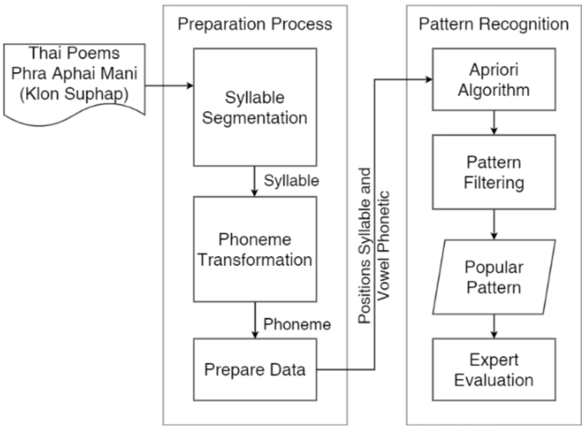
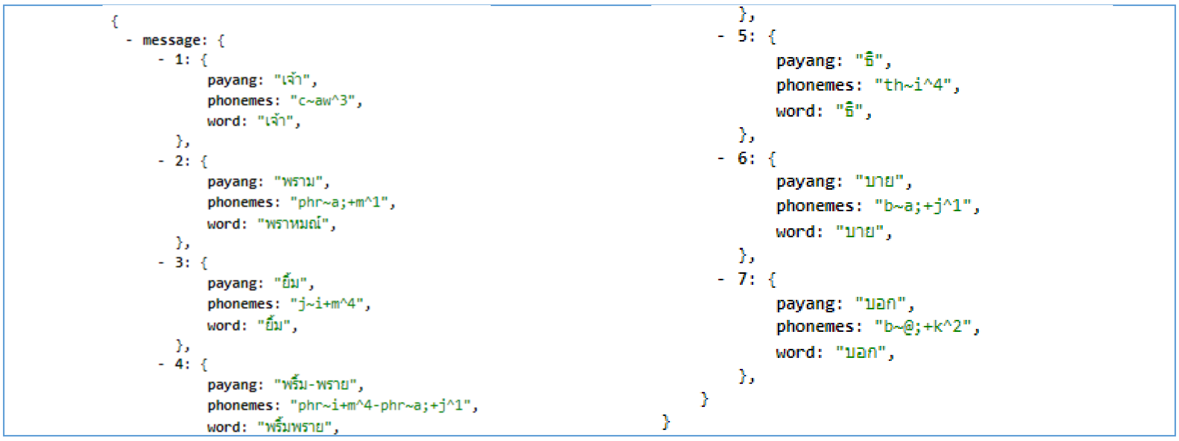


Fig. 5. **Process Overview**

This research used a total of 25 episodes (15,796 sentences) of Sunthorn Phu's fable poem named "Phra Aphai Mani" as the data set. The data was compiled from the Vajirayana digital library.

2.2. Data Preparation

This article used the data set (15,796 sentences) segmented into syllables by Syllable Segmentation API [25]. Then, Syllable segmentation transforms the syllable to Phoneme transformation by Phoneme Transformation API [26] to get the consonant, vowel, and tone. The output from the Syllable Segmentation API, URL is http://orathai-api.herokuapp.com/orathai/sentence, and example is URL http://orathai-api.herokuapp.com/orathai/เจ้าพราหมณ์ยิ้มพริ้มพรายธิบายบอก are shown in Fig. 6.



```
{
  - message: {
    - 1: {
          payang: "เจ้า",
          phonemes: "c~aw^3",
          word: "เจ้า",
      },
    - 2: {
          payang: "พราม",
          phonemes: "phr~a;+m^1",
          word: "พราหมณ์",
      },
    - 3: {
          payang: "ยิ้ม",
          phonemes: "j~i+m^4",
          word: "ยิ้ม",
      },
    - 4: {
          payang: "พริ้ม-พราย",
          phonemes: "phr~i+m^4-phr~a;+j^1",
          word: "พริ้มพราย",
      },
    - 5: {
          payang: "ธิ",
          phonemes: "th~i^4",
          word: "ธิ",
      },
    - 6: {
          payang: "บาย",
          phonemes: "b~a;+j^1",
          word: "บาย",
      },
    - 7: {
          payang: "บอก",
          phonemes: "b~@;+k^2",
          word: "บอก",
      },
  }
}
```

Fig. 6. The example output from the Syllable Segmentation API
(URL: http://orathai-api.herokuapp.com/orathai/เจ้าพราหมณ์ยิ้มพริ้มพรายธิบายบอก)

The Syllable Segmentation API on Table 1 shows that the parameter sentence with its desctiption and example input.

Table 1. Syllable Segmentation API

Parameter	Description	Example input
sentence	Poem Wak. (Wak is a sentence.) The sentence means Thai Poem Klon-Suphap 1 sentence. The sentence in the Thai language has no spaces and only 7 to 9 syllables.	เจ้าพราหมณ์ยิ้มพริ้มพรายธิบายบอก The translated in English is "He smiled to explain."

Syllable Segmentation API will return syllable (*payang*), phoneme (phonemes), and word. We were checking the syllable before the syllable into phoneme transformation. The syllable checking is shown in Fig. 7.

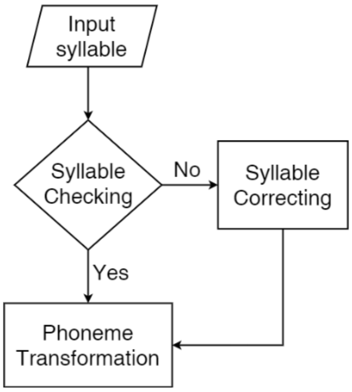


Fig. 7. Syllable Checking

From the structure and the position of the internal rhyme prosody in Fig. 2. The internal rhyme prosodies of Klon-Suphap with 12 rules of Rhyme as in Fig. 8.

R1 relation of $e1$ and $e2$ or ex and $e4$
R2 relation of $e1$ and e2 or ex and ez
R3 relation of ax and fy or ax and fz
R4 relation of $g1$ and g2 or gx and gy
R5 relation of $g1$ and g2 or gx and gz
R6 relation of bx and hy or bx and hz
R7 relation of $i1$ and i2 or ix and iy
R8 relation of $i1$ and $i2$ or ix and iz
R9 relation of cx and jy or cx and jz
R10 relation of k1 and $k2$ or kx and ky
R11 relation of $k1$ and k2 or kx and kz
R12 relation of dx and ly or dx and lz

Fig. 8. The internal rhyme prosodies of Klon-Suphap with 12 rules of Rhyme

2.3. Klon-Suphap Pattern Recognition

This part explains the procedure, which is the Apriori algorithm for this paper. The data set was divided into four groups because Wak has different specifications, including 1) Poem Wak No. 1 3,949 sentences; 2) Poem Wak No. 2 3,949 sentences; 3) Poem Wak No. 3 3,949 sentences; and 4) Poem Wak No. 4 3,949 sentences. The input was structured as a 2-dimensional array. The data within the array consisted of the row data, Wak, the column data, which was the position of the syllable and vowel, converted into the phonetic alphabet. A syllable is a sound uttered at one time, whether that sound has meaning or not. Syllable elements consist of onset, Rhyme, and tone. The structure of the syllable is shown in Fig. 9. A vowel is a sound followed by a consonant, while the vowel's position in the structure of the syllable is shown in Fig. 10.

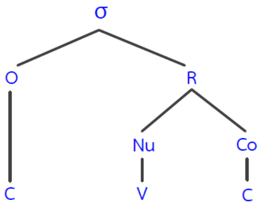


Fig. 9. Structure of Syllable

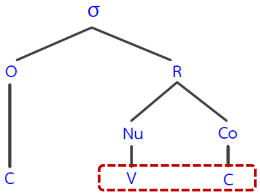


Fig. 10. Position of Vowel in Structure of Syllable

Thai syllable structure can be divided into four types: CV, CCV, CVC, and CCVC. The conversion of vowels into the phonetic alphabet was the process of comparing the sounds to understand the pronunciation of the Thai language. The 44 consonant symbols in the Thai alphabet produce 21 initial consonant sounds and 8 final consonant sounds [27]. There are 24 vowel symbols and 32 vowel phonemes in the Thai language [28]. After phoneme transformation, we used the vowel converted into the phonetic alphabet. The input example, including the syllable position and the vowel that converted into the phonetic alphabet, is shown in Fig. 11.

```
[["(1, 'a;')", "(2, 'x;n')", "(3, 'a;')", "(4, 'a;')", "(5, 'vk')", "(6, 'an')", "(7, 'o;m')", "(8, 'ab')"],
["(1, 'aw')", "(2, 'u;')", "(3, 'a;')", "(4, 'a;')", "(5, 'om')", "(6, '@;')", "(7, 'a')", "(8, 'ad')"],
[......]]
```

Fig. 11. The example of input.

Fig. 12 shows the finding algorithm of the popular assonance patterns. Each data group was put into the Apriori algorithm [25] to find the most frequently found items in the database. Firstly, the set of 1-itemsets would be detected by interpreting the database to count each item's frequency. Then the counting would identify which item met the minimum support. Only the items which counted more than or equal to the minimum support were selected for the next iteration, whereas the others were pruned. For example, the Apriori algorithm setting the minimum support = 50% and the minimum confidence = 80% is shown in Table 2.

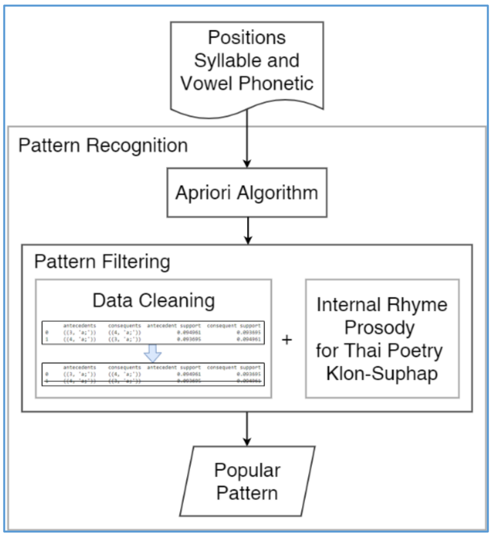
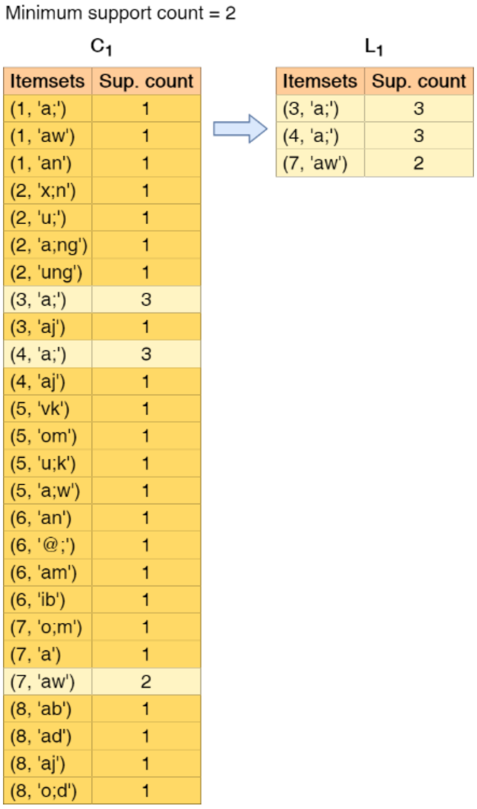

Positions Syllable and Vowel Phonetic
Pattern Recognition
Apriori Algorithm
Pattern Filtering
Data Cleaning
+
Internal Rhyme Prosody for Thai Poetry Klon-Suphap
Popular Pattern

Fig. 12. The finding of the popular assonance patterns

Table 2. Example of syllables and vowels in each Wak

Transaction	Items appearing in the transaction
T1	{(1, 'a;'), (2, 'x;n'), (3, 'a;'), (4, 'a;'), (5, 'vk'), (6, 'an'), (7, 'o;m'), (8, 'ab')}
T2	{(1, 'aw'), (2, 'u;'), (3, 'a;'), (4, 'a;'), (5, 'om'), (6, '@;'), (7, 'a'), (8, 'ad')}
T3	{(1, 'aw'), (2, 'a;ng;'), (3, 'a;'), (4, 'a;'), (5, 'u;k'), (6, 'am'), (7, 'aw'), (8, 'aj')}
T4	{(1, 'an'), (2, 'ung;'), (3, 'aj'), (4, 'aj'), (5, 'a;w'), (6, 'ib'), (7, 'aw'), (8, 'o;d')}

The frequent 1-itemsets (L_1) from the given data was founded based on the candidate 1-itemsets (C_1), then the support count of each 1-itemsets of C_1 would be compared with the minimum support specifying the qualified 1-itemsets, called frequent 1-itemsets (L_1). All those procedures are shown in Fig. 13. From the example, the set minimum support = 50% (50/100)*4 = 2 transactions. Thus, the itemsets qualified for the support criteria must have a greater than or equal to 2. After that, the frequent 2-itemsets (L_2) would be found based on the candidate 2-itemsets (C_2). C_2 would be created from the itemsets from L_1 in Fig. 13. Then both were joined together, as shown in Fig. 14. From the aforementioned step, all frequent itemsets were {(3, 'a;')} {(4, 'a;')} {(7, 'aw')} and {(3, 'a;'), (4, 'a;')}. By determining the correlation of the syllables and the vowel positions in two patterns, the relationship rules could be created from {(3, 'a;'), (4, 'a;')} as (3, 'a;') → (4, 'a;') [75%, 100%], and (4, 'a;') → (3, 'a;') [75%, 100%].



Minimum support count = 2

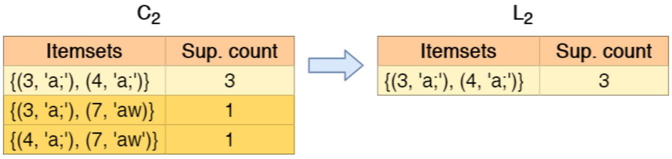
C_1

Itemsets	Sup. count
(1, 'a;')	1
(1, 'aw')	1
(1, 'an')	1
(2, 'x;n')	1
(2, 'u;')	1
(2, 'a;ng')	1
(2, 'ung')	1
(3, 'a;')	3
(3, 'aj')	1
(4, 'a;')	3
(4, 'aj')	1
(5, 'vk')	1
(5, 'om')	1
(5, 'u;k')	1
(5, 'a;w')	1
(6, 'an')	1
(6, '@;')	1
(6, 'am')	1
(6, 'ib')	1
(7, 'o;m')	1
(7, 'a')	1
(7, 'aw')	2
(8, 'ab')	1
(8, 'ad')	1
(8, 'aj')	1
(8, 'o;d')	1

L_1

Itemsets	Sup. count
(3, 'a;')	3
(4, 'a;')	3
(7, 'aw')	2

Fig. 13. Candidate 1-itemsets (C_1) and Frequent 1-itemsets (L_1).



C_2

Itemsets	Sup. count
{(3, 'a;'), (4, 'a;')}	3
{(3, 'a;'), (7, 'aw)}	1
{(4, 'a;'), (7, 'aw')}	1

L_2

Itemsets	Sup. count
{(3, 'a;'), (4, 'a;')}	3

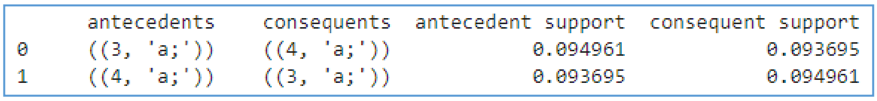
Fig. 14. Candidate 2-itemsets (C_2) and Frequent 2-itemsets (L_2).

The percentages after the relationship rules were minimum support and minimum confidence, which could be calculated from support (A→ B) = P(AUB) and confidence (A→B) = P(B|A) = P(AUB)/P(A). Since the minimum confidence set = 80%, the rule was that each Wak in the Poem had the third syllable sound as vowel a; the fourth syllable sound as vowel a; which occur together in 75% of all transactions. Furthermore, If the third syllable sounded as vowel a; the fourth syllable sounded as vowel a; or if the fourth syllable sounded as vowel a; the third syllable sounded as a; by 100%. From dividing the data into four groups, the research can show the number of relationships patterns in Table 3.

Table 3. Number of patterns relationships

Wak No.	Number of pattern
1	265
2	201
3	220
4	270

The outcome relationship model was cleaned by removing the duplicated vowel pairs in conjunction with the use of internal rhyme prosody to get the vowel pattern of each Wak. The sample of the repeating vowel pairs is shown in Fig. 15, and the number of patterns of the filtered data groups is shown in Table 4.



```
    antecedents    consequents  antecedent support  consequent support
0   ((3, 'a;'))    ((4, 'a;'))            0.094961            0.093695
1   ((4, 'a;'))    ((3, 'a;'))            0.093695            0.094961
```

Fig. 15. The example of the duplicate assonance

Table 4. Number of relation popular pattern

Wak No.	Number of pattern
1	42
2	30
3	46
4	31

In Fig. 15, the vowel syllable No. 3 (a;) and syllable No. 4 (a;), and the vowel syllable No. 4 (a;) and syllable No. 3 (a;) were repeating two pairs of vowels which alternate positions syllable, but had the same meaning. Therefore, one pair had to be removed to reduce the redundancy of the relationship model. After filtering the data, a total of 25 popular pattern vowels of assonance were obtained and divided into 13 short vowels, 10 long vowels, 1 short-long vowel, and 1 long-short vowel—as shown in Table 5.

Table 5. Popular vowel pattern of assonance

Vowel	Popular vowel pattern of assonance
Short vowel	อะ (a), อัก (ak), อัด (ad), อับ (ab), อัน (an), อัง (ang), อำ (am), ไอ/ใอ (aj), เอา (aw), อิด (id), อุด (ud), อน (on), อง (ong)
Long vowel	อา (a;), อาน (a;n), อาง (a;ng), อาม (a;m), อาย (a;j), อี (i;), อู(u;), แอง (x;ng), ออน (@;n), ออง (@;ng)
Short -Long vowel	อะ (a), อา (a;)
Long -Short vowel	อา (a;), อะ (a)

From Table 4 and Table 5, the popular vowel pattern of each Wak were divided into 4 Wak including 1) Wak No. 1 42 patterns; 2) Wak No. 2 30 patterns; 3) Wak No. 3 45 patterns; and 4) Wak No. 4 29 patterns. The rhyme style in Wak No. 1 and Wak No. 3, also Wak No. 2 and Wak No. 4 have the same characteristic. According to Fig. 2, a similar number of popular assonance patterns could be found.

3. Results and Discussion

Sound format extraction, the most common of the popular pattern vowel in Wak No. 1 was $a;$ Wak No. 2 was the pair of a vowel $a - a;$ Wak No. 3 was $a;$ and Wak No. 4 was the pair of vowels $a - a;$. Showing the most common popular pattern in each Wak in Table 6. From Table 6, The syllable pairs of the popular vowel pattern can be seen in Table 7.

Table 6. The most number of popular pattern in each Wak

Wak No.	Number of popular pattern (vowel)	
	a;	a-a;
1	6	-
2	-	6
3	6	-
4	-	6

Table 7. Syllable pairs of the popular pattern

Pattern	Syllable Pairs			
	Wak No.1	Wak No.2	Wak No.3	Wak No.4
	a;	a-a;	a;	a-a;
1	3, 4	4, 6	3, 4	4, 6
2	5, 7	5, 6	5, 7	5, 6
3	6, 8	7, 9	6, 8	7, 8
4	5, 6	7, 8	5, 6	6, 8
5	2, 3	6, 8	2, 3	2, 3
6	6, 7	5, 7	6, 7	6, 7

From Table 7, the syllable pairs of the popular pattern vowel found in each Wak are described as in Fig. 16.

Wak No. 1 has popular vowel rhyme a; was 6 patterns

1) syllables rhyme is 3 and 4
2) syllables rhyme is 5 and 7
3) syllables rhyme is 6 and 8
4) syllables rhyme is 5 and 6
6) syllables rhyme is 6 and 7
5) syllables rhyme is 2 and 3

Wak No. 2 popular vowel rhyme a-a; is 6 patterns

1) syllables rhyme is 4 and 6
2) syllables rhyme is 5 and 6
3) syllables rhyme is 7 and 9
4) syllables rhyme is 7 and 8
5) syllables rhyme is 6 and 8
6) syllables rhyme is 5 and 7

Wak No. 3 popular vowel rhyme a; is 6 patterns

1) syllables rhyme is 3 and 4
2) syllables rhyme is 5 and 7
3) syllables rhyme is 6 and 8
4) syllables rhyme is 5 and 6
5) syllables rhyme is 2 and 3
6) syllables rhyme is 6 and 7

Wak No. 4 popular vowel rhyme a-a; is 6 patterns

1) syllables rhyme is 4 and 6
2) syllables rhyme is 5 and 6
3) syllables rhyme is 7 and 8
4) syllables rhyme is 6 and 8
5) syllables rhyme is 2 and 3
6) syllables rhyme is 6 and 7

Fig. 16. the syllable pairs of the popular pattern vowel found in each Wak are

The examples of verse Wak that have a pair of popular vowel patterns of each Wak are shown in Table 8. The examples of Wak No. 2 and Wak No. 4, respectively. The vowel rhyme a-a; in the fourth and sixth syllable rhyme is one word. Many vowels were found as a-a; due to the use of words in the story. Phra Aphai Mani is a cloak and sword drama in which the words such as "กษัตรา", which translated into English as "The King", and "พระบิดา" which is translated into English as "The father" are words generated from the combination of a and a;. Those words do not currently use in the modern Poem. Therefore, the second popular vowel pattern in Wak No. 2 and Wak No. 4 were found a vowel a; and aj shown in Table 9.

Table 8. Verse of Wak that have a pair of popular vowel pattern of each Wak

Wak No.	Assonance (vowel)	Syllable Pairs	Wak (EN)	Wak (TH)	Syllable	Phoneme
1	a;	3, 4	The boy pleaded with sweet words.	กุมาราว่าวอนด้วยอ่อนหวาน	กุ-มา-รา-ว่า-วอน-ด้วย-อ่อน-หวาน	k~u^2-m~a;^1-r~a;^1-w~a;^3-w~@;+n^1-d~u;aj^3-?~@;+n^2-w~a;+n^5
2	a-a;	4, 6	To say goodbye to the king.	ทูลลาจอมกษัตราอัชฌาสัย	ทูล-ลา-จอม-กะ-ษัต-ตรา-อัด-ชา- ไส	th~u;+n^1-l~a;^1-c~@;+m^1-k~a^2-s~a+d^2-tr~a;^1-?~a+d^2-ch~a;^1-s~aj^5
3	a;	3, 4	Kindness to the Brahmin.	กรุณาปรานีกับชีพราหมณ์	กะ-รุ-ณา-ปรา-นี-กับ-ชี-พราม	k~a^2-r~u^4-n~a;^1-pr~a;^1-n~i;^1-k~a+b^2-ch~i;^1-phr~a;+m^1
4	a-a;	4, 6	Parents rejoice.	ให้ชื่นจิตพระบิดาแลมารดร	ให้-ชื่น-จิต-พระ-บิ-ดา-แล-มาร-ดร	h~aj^3-ch~v;+n^3-c~i+d^2-phr~a^4-b~i^2-d~a;^1-l~x;^1-m~a;+n^1-d~@;+n^1

Table 9. The number of second popular pattern (vowel) in Wak No. 2 and Wak No. 4

Wak No.	Number of popular pattern (vowel)	
	a;	aj
2	4	4
4	5	5

Table 9 could produce the syllable pairs of the popular vowel pattern that can be seen in Table 10. The syllable pairs of the popular pattern vowel found in each Wak are described as in Fig. 17.

Wak No. 2 popular vowels rhyme *a;* is 4 patterns
1) syllables rhyme is 5 and 7
2) syllables rhyme is 6 and 8
3) syllables rhyme is 6 and 7
4) syllables rhyme is 5 and 6

Wak No. 2 popular vowel rhyme *aj* is 4 patterns
1) syllables rhyme is 5 and 7
2) syllables rhyme is 5 and 6
3) syllables rhyme is 6 and 8
4) syllables rhyme is 6 and 7

Wak No. 4 popular vowel rhyme *a;* is 5 patterns
1) syllables rhyme is 5 and 7
2) syllables rhyme is 6 and 8
3) syllables rhyme is 3 and 5
4) syllables rhyme is 5 and 6
5) syllables rhyme is 6 and 7

Wak No. 4 popular vowel rhyme *aj* is 5 patterns
1) syllables rhyme is 5 and 7
2) syllables rhyme is 6 and 8
3) syllables rhyme is 5 and 6
4) syllables rhyme is 6 and 7
5) syllables rhyme is 2 and 3

Fig. 17. The syllable pairs of the popular pattern vowel found in each Wak

Table 10. Syllable pairs of the second popular pattern

Pattern	Syllable Pairs			
	Wak No. 2		Wak No. 4	
	a;	aj	a;	aj
1	5, 7	5, 7	5, 7	5, 7
2	6, 8	5, 6	6, 8	6, 8
3	6, 7	6, 8	3, 5	5, 6
4	5, 6	6, 7	5, 6	6, 7
5	-	-	6, 7	2, 3

The examples of verse Wak that have a pair of popular vowel patterns of each Wak are shown in Table 11.

Table 11. The example of Wak No. 2 and No. 4

Wak	Assonance (vowel)	Syllable Pairs	Wak (EN)	Wak (TH)	Syllable	Phoneme
No. 2	a;	5, 7	The father speaks badly because he was angry.	บิดาตรัสโกร**ธา**ไม่**ปรา**ศรัย	บิ-ดา-ตรัส-โกร-**ธา**-ไม่-**ปรา**-ไส	b~i^2-d~a;^1-tr~a+d^2-kr~o;^1-**th~a;^1**-m~aj^3-**pr~a;^1**-s~aj^5
	aj	5, 7	Hearing the sound was passionate.	ยินสำเนียงถึง**ไหน**ก็**ไหล**หลง	ยิน-สำ-เนียง-ถึง-**ไหน**-ก็-**ไหล**-หลง	j~i+n^1-s~am^5-n~i;a+ng^1-th~v+ng^5-**n~aj^5**-k~@^3-**l~aj^5**-l~o+ng^5
No. 4	a;	5, 7	The people of the city of joy enduring.	ชาวบุรีหรร**ษา**ส**ถา**วร	ชาว-บุ-รี-หัน-**ษา**-สะ-**ถา**-วร	ch~a;+w^1-b~u^2-r~i;^1-h~a+n^5-**s~a;^5**-s~a^2-**th~a;^5**-w~@;+n^1
	aj	5, 7	In the evening, he invites his son to go to bed.	ครั้นค่ำชวนหน่อ**ไท**เข้า**ไส**ยา	ครั้น-ค่ำ-ชวน-หน่อ-**ไท**-เข้า-**ไส**-ยา	khr~a+n^4-kh~am^3-ch~u;a+n^1-n~@;^2-**th~aj^1**-kh~aw^3-**s~aj^5**-j~a;^1

The questionnaire for experts to assess their opinions on the melodiousness of the rhyme patterns extracted from the story of Phra Aphai Mani was developed. The opinions of 3 experts evaluated an example of 18 Bots totals of 54 Bots. There were 216 Waks, divided into two pattern formats: 1) melodious 144 waks and 2) inferior melodious 72 waks.

Melodious pattern No. 1 is a verse in the bots where the pattern extraction tool found all assonances and found a total of 25 vowel rhymes. See at Table 3. The Assessment of melodiousness by experts compared with the system. In which the Poem has the melodious pattern No. 1 was a melodious poem. All three experts agreed. Second, Melodious pattern No. 2 is a verse in the bots where the pattern extraction tool found all assonances, but some waks may not have the sound that matched the melodious pattern No. 1. The Assessment of melodiousness by experts compared with the system. In which the Poem has the melodious pattern No. 2 was a melodious poem. All three experts agreed.

In addition, The inferior melodious pattern is a verse in the bots where the pattern extraction tool found some waks that did not have a vowel. The Assessment of melodiousness by experts compared with the system. In which the Poem has the inferior melodious pattern was a poem that was not melodious. Two experts agreed, and one expert disagreed. From the Assessment of melodiousness by experts compared with the system and the examination of the melodiousness efficiency of the Poem which was extracted by the tool, There were 144 Waks used in the examination. The results of the melodiousness are shown in Table 12. The consistency score from 3 experts shown is 95.30%

Table 12. Melodiousnessassessment by experts who agree on the melodiousness that the tool extracted

Expert No.	Agree	Disagree
1	136	8
2	140	4
3	136	8
Consistency score	**0.953**	

4. Conclusion

This research presents the extraction of the beautiful sound patterns from Sunthorn Phu's poem "Phra Aphai Mani" which he frequently used to compose other poems. From the evaluation of the popular vowel pattern and the Assessment of melodiousness by experts, the experts agreed that the melodious pattern No. 1 is melodious. This research illustrates that the melodiousness of a poem depends not only on assonance but also on alliteration. A poem that has both assonance and alliteration will be more even melodious. The extraction of the melodious sound patterns focuses only on assonance, without considering alliteration, an important element for melodiousness evaluation. Therefore, the development of tools to investigate alliteration will help improve the efficiency of the melodiousness pattern extraction.

Acknowledgment

The authors thank the Thai Contemporary Poets Association for knowledge support. And also thank the advisor for their support, appreciated the encouragement.

Declarations

Author contribution. All authors contributed equally to the main contributor to this paper. All authors read and approved the final paper.

Funding statement. None of the authors have received any funding or grants from any institution or funding body for the research.

Conflict of interest. The authors declare no conflict of interest.

Additional information. No additional information is available for this paper.

References

[1] S. Waijanya and N. Promrit, “The Poet Identification Using Convolutional Neural Networks,” 2018, pp. 179–187. doi: 10.1007/978-3-319-60663-7_17

[2] A. Meesing, “Book Review-The Journey to Petchburi: A Poem by Sunthorn Phu,” *ABAC J.*, vol. 36, no. 2, pp. 135–139, 2016. Available at: Google Scholar.

[3] G. Li and J. Li, “Research on Sentiment Classification for Tang Poetry based on TF-IDF and FP-Growth,” in *2018 IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, 2018, pp. 630–634, doi: 10.1109/IAEAC.2018.8577715.

[4] R. Rajan and A. A. Raju, “Deep Neural Network Based Poetic Meter Classification Using Musical Texture Feature Fusion,” in *2019 27th European Signal Processing Conference (EUSIPCO)*, 2019, pp. 1–5, doi: 10.23919/EUSIPCO.2019.8902998.

[5] N. Promrit and S. Waijanya, “Convolutional Neural Networks for Thai Poem Classification,” 2017, pp. 449–456. doi: 10.1007/978-3-319-59072-1_53

[6] J. Kaur and J. R. Saini, “Punjabi poetry classification: the test of 10 machine learning algorithms,” in *Proceedings of the 9th International Conference on Machine Learning and Computing*, 2017, pp. 1–5. doi: 10.1145/3055635.3056589

[7] A. M. S. Rahma, M. A. H. Alrawi, and N. A. G., “A Novel Coding and Discremenation (CODIS) Algorithm to Extract Features from Arabic Texts to Discriminate Arabic Poems,” in *2018 1st Annual International Conference on Information and Sciences (AiCIS)*, 2018, pp. 117–128, doi: 10.1109/AiCIS.2018.00033.

[8] W.-C. Yeh, Y.-C. Chang, Y.-H. Li, and W.-C. Chang, “Rhyming Knowledge-Aware Deep Neural Network for Chinese Poetry Generation,” in *2019 International Conference on Machine Learning and Cybernetics (ICMLC)*, 2019, pp. 1–6, doi: 10.1109/ICMLC48188.2019.8949208.

[9] D. Liu, J. Lv, and Y. Li, “Generating Style-Specific Chinese Tang Poetry With a Simple Actor-Critic Model,” *IEEE Trans. Emerg. Top. Comput. Intell.*, vol. 3, no. 4, pp. 313–321, Aug. 2019, doi: 10.1109/TETCI.2018.2870125.

[10] Y. Zhang, Y. Li, X. Wei, and L. Jia, “Adaptive spatio-temporal graph convolutional neural network for remaining useful life estimation,” in *2020 International joint conference on neural networks (IJCNN)*, 2020, pp. 1–7, doi: 10.1109/IJCNN48605.2020.

[11] N. Promrit, S. Waijanya, and K. Thaweesith, “The Evaluation of Thai Poem’s Content Consistency using Siamese Network,” in *Proceedings of the 2019 3rd International Conference on Natural Language Processing and Information Retrieval*, 2019, pp. 115–120, doi: 10.1145/3342827.3342855.

[12] S. Waijanya and A. Mingkhwan, “Thai poetry translation to English with backward translation evaluation,” in *Ninth International Conference on Digital Information Management (ICDIM 2014)*, 2014, pp. 248–253, doi: 10.1109/ICDIM.2014.6991425.

[13] S. Seljan, I. Dunder, and M. Pavlovski, “Human Quality Evaluation of Machine-Translated Poetry,” in *2020 43rd International Convention on Information, Communication and Electronic Technology (MIPRO)*, 2020, pp. 1040–1045, doi: 10.23919/MIPRO48935.2020.9245436.

[14] I. Dunder, S. Seljan, and M. Pavlovski, “Automatic Machine Translation of Poetry and a Low-Resource Language Pair,” in *2020 43rd International Convention on Information, Communication and Electronic Technology (MIPRO)*, 2020, pp. 1034–1039, doi: 10.23919/MIPRO48935.2020.9245342.

[15] O. Khongtum, N. Promrit, and S. Waijanya, “The Entity Recognition of Thai Poem Compose by Sunthorn Phu by Using the Bidirectional Long Short Term Memory Technique,” 2019, pp. 97–108. doi: 10.1007/978-3-030-33709-4_9

[16] A. Mittmann, A. von Wangenheim, and A. L. dos Santos, “A Multi-level Visualization Scheme for Poetry,” in *2016 20th International Conference Information Visualisation (IV)*, 2016, pp. 312–317, doi: 10.1109/IV.2016.64.

[17] M. A. Ahmed and S. Trausan-Matu, “Using natural language processing for analyzing Arabic poetry rhythm,” in *2017 16th RoEduNet Conference: Networking in Education and Research (RoEduNet)*, 2017, pp. 1–5, doi: 10.1109/ROEDUNET.2017.8123759.

[18] W. Menninghaus, V. Wagner, E. Wassiliwizky, T. Jacobsen, and C. A. Knoop, "The emotional and aesthetic powers of parallelistic diction," *Poetics*, vol. 63, pp. 47–59, 2017. doi: 10.1016/j.poetic.2016.12.001

[19] S. Harikumar and D. U. Dilipkumar, "Apriori algorithm for association rule mining in high dimensional data," in *2016 International Conference on Data Science and Engineering (ICDSE)*, 2016, pp. 1–6, doi: 10.1109/ICDSE.2016.7823952.

[20] B. Li, Q. Ji, Z. Mi, Y. Yang, and Y. Guo, "An Improved Apriori Algorithm Applied to Mining Ancient Chinese Poems," in *2018 International Conference on Computer, Information and Telecommunication Systems (CITS)*, 2018, pp. 1–5, doi: 10.1109/CITS.2018.8440132.

[21] J. Liu, F. Wu, C. Wu, Y. Huang, and X. Xie, "Neural Chinese word segmentation with dictionary," *Neurocomputing*, vol. 338, pp. 46–54, Apr. 2019, doi: 10.1016/j.neucom.2019.01.085.

[22] Y. Wei, H. Wang, J. Zhao, Y. Liu, Y. Zhang, and B. Wu, "GeLaiGeLai: A visual platform for analysis of Classical Chinese Poetry based on Knowledge Graph," in *2020 IEEE International Conference on Knowledge Graph (ICKG)*, 2020, pp. 513–520, doi: 10.1109/ICBK50248.2020.00078.

[23] S. Waijanya and A. Mingkhwan, "The Evaluations of Thai Poetry Translator to English with Prosody Keeping.," *J. Digit. Inf. Manag.*, vol. 12, no. 6, pp. 357–368, 2014. Available at: Google Scholar.

[24] D. D. McCracken and E. D. Reilly, "Backus-naur form (bnf)," in *Encyclopedia of Computer Science*, 2003, pp. 129–131. Available at: Google Scholar.

[25] J. Yabing, "Research of an Improved Apriori Algorithm in Data Mining Association Rules," *Int. J. Comput. Commun. Eng.*, pp. 25–27, 2013, doi: 10.7763/IJCCE.2013.V2.128.

[26] H. Turner, G. Lovisotto, and I. Martinovic, "Attacking speaker recognition systems with phoneme morphing," in *European Symposium on Research in Computer Security*, 2019, pp. 471–492. doi: 10.1007/978-3-030-29959-0_23

[27] Sunee Leelapornpinit, "A comparative study of thai and chinese phonology for the use of basic thai language teaching as a foreign language," *Suthiparithat J.*, vol. 30, no. 93, pp. 33–46, 2016. Available at: Suthiparithat Journal

[28] H. Winskel and T. Ratitamkul, "The Initial Functional Unit When Naming Words and Pseudowords in Thai: Evidence from Masked Priming," *J. Psycholinguist. Res.*, vol. 49, no. 2, pp. 275–290, 2020. doi: 10.1007/s10936-020-09687-7