Computational Intelligence, Volume 23, Number 1, 2007

KEYWORD EXTRACTION STRATEGY FOR ITEM BANKS TEXT CATEGORIZATION

ATORN NUNTIYAGUL

Institute for Innovation and Development of Learning Process, Mahidol University, Thailand

KANLAYA NARUEDOMKUL

Department of Mathematics, Faculty of Science, Mahidol University, Thailand

NICK CERCONE

Faculty of Computer Science, Dalhousie University, Canada

DAMRAS WONGSAWANG

Department of Computer Science, Faculty of Science, Mahidol University, Thailand

We proposed a feature selection approach, *Patterned Keyword in Phrase* (*PKIP*), to text categorization for item banks. The item bank is a collection of textual question items that are short sentences. Each sentence does not contain enough relevant words for directly categorizing by the traditional approaches such as “bag-of-words.” Therefore, PKIP was designed to categorize such question item using only available keywords and their patterns. PKIP identifies the appropriate keywords by computing the weight of all words. In this paper, two keyword selection strategies are suggested to ensure the categorization accuracy of PKIP. PKIP was implemented and tested with the item bank of Thai high primary mathematics questions. The test results have proved that PKIP is able to categorize the question items correctly and the two keyword selection strategies can extract the very informative keywords.

Key words: item bank, feature selection, patterned keywords in phrase, text categorization, keyword extraction.

1. INTRODUCTION

In general, an item bank is a collection of items related by some common features, such as similar content or common purpose. In educational area, an item bank is frequently referred to a collection of question items that are stored in the database and can be retrieved for a test or exam by users.

An item bank is a valuable tool not only for students but also for teachers. For students, an item bank is a resource for practicing that can help improving their learning ability. For teachers, an item bank can be used to assess the students’ knowledge and the development of intellectual skill in the cognitive domain (Wiggins 1998). Therefore, the demand for item banks has, not surprisingly, increased especially for the subjects that require a lot of practicing, i.e., Mathematics, Physics, and General Chemistry.

Currently, the available question item banks for each subject were developed separately. Tests and exams were collected over time from several sources including instructors, students, experts, competitions, textbooks, and educational publishing companies. Here, the item banks can be divided into three groups according to their management methodologies. In the first group, the question items were not systematically organized. In the second group, the items were arranged in chronological order. To use the item banks from these two groups, users must spend a lot of time and effort trying to retrieve what they are looking for and may end up retrieving nothing. In the third group, the items were organized by experts. In this item bank, users are allowed to retrieve the items more conveniently; however, it is not only time consuming but also costs a lot of money. In addition, the size of the item banks is increasing every day thus it can hardly avoid human errors.

Each question item consists of phrases, clauses and/or short sentences. Some questions are objective tests that are multiple-choice, true-false, and matching items. An item mostly

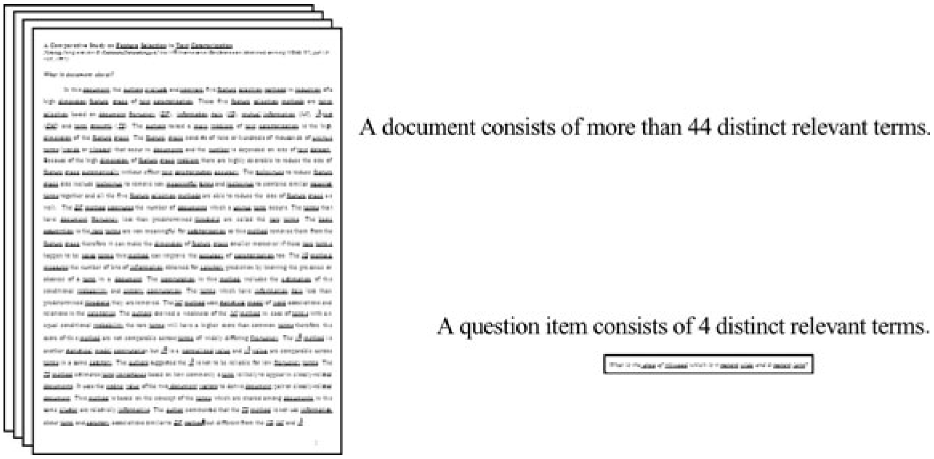

A document consists of more than 44 distinct relevant terms.
A question item consists of 4 distinct relevant terms.

FIGURE 1. Pictorials of a document and a question item.

contains less than 50 words whereas each document (in text document collections) consists of many hundred words. Figure 1 illustrates examples of a document against a question item. The underlined words are relevant terms that can be used in categorization process. Note that a question item contains a lot less relevant terms compared to a document. This causes the existing text categorization techniques become infeasible to provide sufficiently accurate results when used directly with question item banks because these techniques were designed to support the large text documents such as Reuter collection, OHSUMED collection, AP newswire collection, 20 Newsgroups collection, and large web pages.

To be able to categorize the question items correctly even though they do not have enough of keywords, PKIP is proposed. PKIP was designed to manage question item banks systematically and automatically, to allow users to store, reuse, and retrieve items with ease.

2. RELATED WORK

In text categorization (or text classification), a number of researches focused on textual data representation because it strongly affects the efficiency and accuracy of categorization. To completely represent the meaning of the text, the system should be able to indicate the importance of the word or term. A number of factors used in representations were suggested, i.e., term frequency (TF), inverse document frequency (IDF), and term frequency with inverse document frequency (TFIDF).

TF is a frequency of the terms occurrences across the document. It is based on the concept that the more often a term occurs in a document, the more important it is in describing that document. IDF means that the more documents a term appears in, the less important the term is. TFIDF suggested that the importance of the word increases proportionally to the number of times a word appears in the document but is offset by how common the word is in all of the documents in the collection.

TFIDF was introduced by Salton and Buckley (1988). It is the most widely used technique in information retrieval and text mining. Though it requires some fine tunings to increase efficiency and accuracy, many researchers have adopted the TFIDF technique for their data. In

2002, Jing, Huang, and Shi (2002) proposed a new TFIDF-based feature selection approach to improve the accuracy of text mining. They use a feature vector as a document representation that is to take a document as a set of term sequences that include term and term weight. The term weight, W_i, of the term t_i in document d is conventionally calculated from:

$$W_i = TF(t_i, d) * IDF(t_i) \quad (1 \leq i \leq n) \tag{1}$$

W_i signifies that the word t_i is an important indexing term if it presents the highest occurrences frequently in the document d relative to its appearances in the overall document collection.

In Jing's work, the weighting function was modified into:

$$W_i = TF(t) * MutualInfoTxt(t, c_i), \tag{2}$$

where $TF(t)$ denotes the term frequency in the document d, and $MutualInfoTxt(t, c_i)$ represents mutual information (MI) of one word in all class sets, and

$$MutualInfoTxt(t, c_i) = \sum_i P(c_i) \log \frac{P(t \mid c_i)}{P(t)} \quad (1 \leq i \leq n). \tag{3}$$

MI for text is defined as mutual information for a fixed feature value of word t occurred in category c_i averaged over the number of words t occurred in all classes.

With this feature representation, the classification accuracy by Naïve Bayes classifier is improved about 12%.

Another attempt to use an improved automatic feature selection method in conjunction with conventional classifiers was proposed by Ghanem et al. (2002). Their feature selection is based on the number of times a keyword combination appears in the document. This provides more accurate classification results than other approaches that use keyword-based feature.

Sakurai and Suyama (2004) studied the relation between keywords. Their proposed method decomposes the textual data into word sets using lexical analysis. The training examples were generated from both key phase relations extracted from the word sets (using key phase patterns) and from text classes given by the user. Key phase relation rules are generated from the example by using a fuzzy inductive learning algorithm. This method can apply to any textual data that require word segmentation, i.e., Japanese and some Asian texts.

Some text categorizations were applied to another kind of data set apart from the news group data set such as survey coding. Survey coding is the task of taking a set of textual responses to open-ended survey questions and assigning each response to a predefined coding category as in Giorgetti and Sebastiani (2003). They suggested that an automatic survey coding can be considered as a multiclass text categorization. Their proposed technique significantly outperforms the dictionary-based techniques that are conventional approaches to automated survey coding.

Li and Roth (2002) are interested in machine learning approaches to question categorization. They argued that local features of the questions are insufficient to deliver the accurate classification result. Therefore, they developed a hierarchical classifier that was guided by semantic of answers' types together with Text REtrieval Conference (TREC) questions (details are provided at http://trec.nist.gov). Although their system illustrated the acceptable classification performance, it is based on semantic analysis technique and the features were constructed semiautomatically.

Zhang and Sun Lee (2003) presented their contribution to automatic question classification with two kinds of features and five machine learning algorithms in comparison. Their experiment results illustrated that the support vector machines algorithm, SVM, outperforms the other four algorithms and bag-of-words feature is slightly better than n-gram feature.

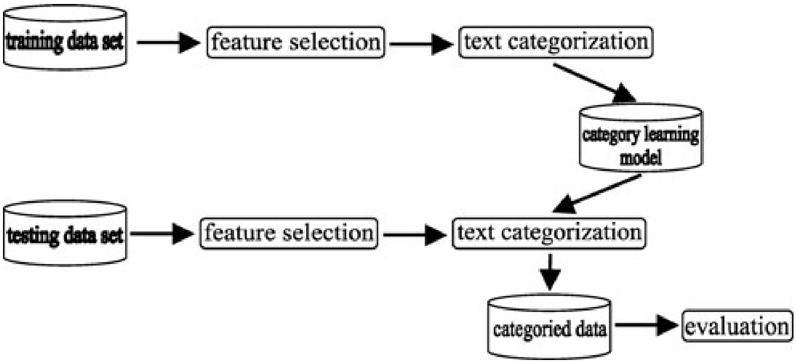


FIGURE 2. Overall system architecture.

3. OVERALL SYSTEM ARCHITECTURE

The architecture of PKIP, the item bank categorization system, is illustrated in Figure 2. PKIP begins with a training phase to teach the system how to categorize each item into a corresponding predefined category. This supervised machine learning based approach produces the item bank categorization in a two-step process.

In the first step, the system induces categorization rules (or category learning model) from the training data with predefined categories. These categorization rules are repeatedly used in the second step to categorize incoming question items. In each step, the input textual data are transformed and some specific features are selected before categorization, details are described in the next two sections.

4. FEATURE SELECTION

The feature selection process consists of five main parts including text representation, preprocessing, keyword extraction, patterned keyword generation, and vector space model generation as shown in Figure 3.

4.1. Text Representation

Before performing feature selection, a textual data must be represented into a computational data form. There are many different approaches to represent textual data. Joachims (2002) suggested that the text representation approaches can be classified into four groups according to the level of text analysis.

4.1.1. Subword. This approach uses sequences of consecutive characters as indexing terms, *n*-grams approach is for example. *N* is a specified number of consecutive characters. The advantage of this approach is that it can be used with any kind of textual data. Furthermore, it is language independent and can be used without concerning the meaning of text.

4.1.2. Word. This approach is widely used to represent textual data because word is a basic unit of text that is meaningful and easy to consider. This approach considers textual data as a sequence of words without considering context and neglects the information of words

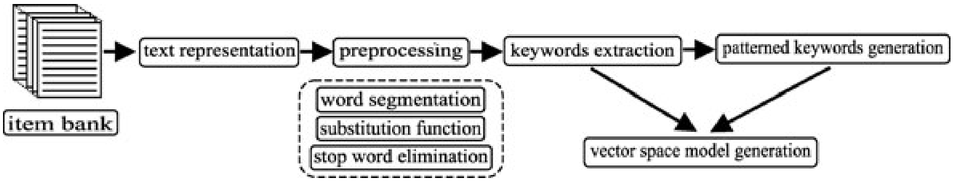


FIGURE 3. Feature selection process.

ordering. “Bag-of-words” is an example approach. One advantage of the “bag-of-words” approach is its simplicity.

4.1.3. Multiword. This approach considers a group of words that together have a particular meaning. It uses a group of words that includes syntactic information as indexing term so it is known as the *Syntactic Phrase Indexing* approach. The advantage of multiword representation approach is the reduction of indexing term number and gain more correct meaning from the document.

4.1.4. Semantic. Actually, to be able to capture the meaning of the context is the best way to represent the document. However, in doing so, we need some intelligent being like human to complete this kind of task. Many researches studied for extracting the semantic from the document automatically by the machine. However, this approach is still a challenge topic of text representation for research.

Among these approaches, the bag-of-words approach is widely adopted for most works in text categorization due to its simplicity and effectiveness in information retrieval and text categorization (Lewis 1992; Joachims 2002; Zhang and Sun Lee 2003). It was adopted for PKIP as well. The number of times the word occurs across the document is taken as the value of attribute. A multidimension vector is formed to represent the document.

It is a fact that the data representation strongly affects accuracy of categorization results. The conventional textual data representation method depends on the number of words (or terms) occur across a document after preprocessing processes (stop word elimination and word stemming). These representation methods are based on the belief that terms that have a higher number of occurrences in a document are relevant to the document category. Examples of these methods include TF, TFIDF (Joachims 1997), and so on.

As we mentioned above, the characteristics of an item bank data set is the lack of relevant words (keywords) in each item, unlike a document. Each keyword always occurs once or twice in an item, like the irrelevant words. Sometimes, occurrence frequency of relevant words is less than that of irrelevant words. In our initial experiment, the question items are in Thai language, the language with no word boundary. An example of a question item in Thai high primary mathematics and its corresponding in English are shown below.

Example 1:

Thai: “จง|หา|พื้นที่|ของ|กระเบื้อง|รูป|สี่เหลี่ยมจัตุรัส|ซึ่ง|มี|ด้าน|ยาว|8|นิ้ว”
find area tile square side inch

English: *“Find the area of a square tile which has 8-inch sides”*

This item in Example 1 is supposed to be categorized into "*square*" and "*area of quadrilateral*" categories. The relevant keywords are "*area*" and "*square*" that occur once each. The irrelevant words, i.e., "*find,*" "*tile,*" "*inch,*" "*side*" occur once each as well.

Example 2:

Thai: "ไม้อัด|แผ่น|หนึ่ง|กว้าง|4|เมตร|ยาว|3|เมตร|จะ|มี|พื้นที่|เท่าไร"

Plywood *wide* *meter* *long* *meter* *area*

English: "*What is the area of plywood that is 4-meters wide and 3-meters long?*"

This item, in Example 2, is categorized into "*area of quadrilateral*" category. The relevant keywords, "*area,*" "*wide,*" and "*long,*" occur once each which is less than the number of occurrences of the irrelevant word "*meters.*"

4.2. Preprocessing

Preprocessing includes *word segmentation*, *substitution functions*, *word stemming,* and *stop word elimination.* Which preprocessing is required depends on the nature of language of the input textual data.

Word segmentation is required for textual data that are in Thai or in any language with no word boundary. The words are not delimited by a space, unlike English. Currently, a number of Thai word segmentation methodologies are available.

Substitution functions are applied if the input textual data contain numbers and abbreviations. A number will be replaced with the corresponding text form whereas an abbreviation will be substituted with its full word form.

Word stemming is required for any language, i.e., English, which has inflection resulting from tenses, verb agreement, and plurality. Word stemming applies on words to remove the inflection part. However, this preprocess does not apply on Thai language because Thai has no word inflection.

Stop word elimination is applied to reduce irrelevant features, to help the methods to perform better. Stop words are words that from nonlinguistic view do not carry information; they have mainly functional role. Stop words cannot be used to identify item categorization. The criteria used in determining stop words depend on languages and data set applications. In Thai, stop word list can be constructed from prepositions and conjunctions and can be determined by using statistics of word usage for our data set.

4.3. Keywords Extraction

For the Thai item bank data set, the keywords are words or terms that have a high number of occurrences in the training data set of each category after eliminating all stop words. The process does not use term frequency in a document (tf_d) or in an item for this case because the "lack of terms" characteristic of the item bank. In contrast, the process uses term frequency of a group of items in the same category (tf_c) and the process still uses inverse document frequency (idf) for weighting the terms too.

The weight of term t_k in a group of items in the same category c_j is given by $w_{j,k}$ and we defined weighting function $tf_c idf$ as

$$tf_c idf\,(t_k, c_j) = \#(t_k, c_j) * idf, \tag{4}$$

where

$1 \leq k \leq$ number of total words in category c_j
$1 \leq j \leq$ number of total categories in data set
$\#(t_k,c_j)$ is the number of occurrences of term t_k in a group of items in the same category c_j

and idf = inverse document frequency

$$idf = \left(\frac{T_r}{\#(t_k)}\right). \tag{5}$$

(Note: in this case document = item)
Where T_r *is the number of all items (in every category) in the training set.*

$\#t_k$ *is the number of items (in every category) in the training set in which term* t_k *occurs at least once.*

Therefore, we obtain the weighting function $tf_c idf$ as:

$$tf_c idf(t_k, c_j) = \#(t_k, c_j) * \log\left(\frac{T_r}{\#(t_k)}\right) \tag{6}$$

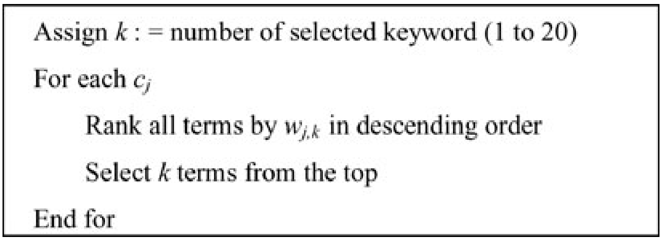
Finally, the weighting function $tf_c idf$ is normalized by cosine normalization and then the weight of term t_k, $w_{j,k}$ can be presented as:

$$w_{j,k} = \frac{tf_c idf(t_k, c_j)}{\sqrt{\sum_{s=1}^{T} (tf_c\, idf(t_s, c_j))^2}}, \tag{7}$$

where T is the set of all terms that occur in T_r.

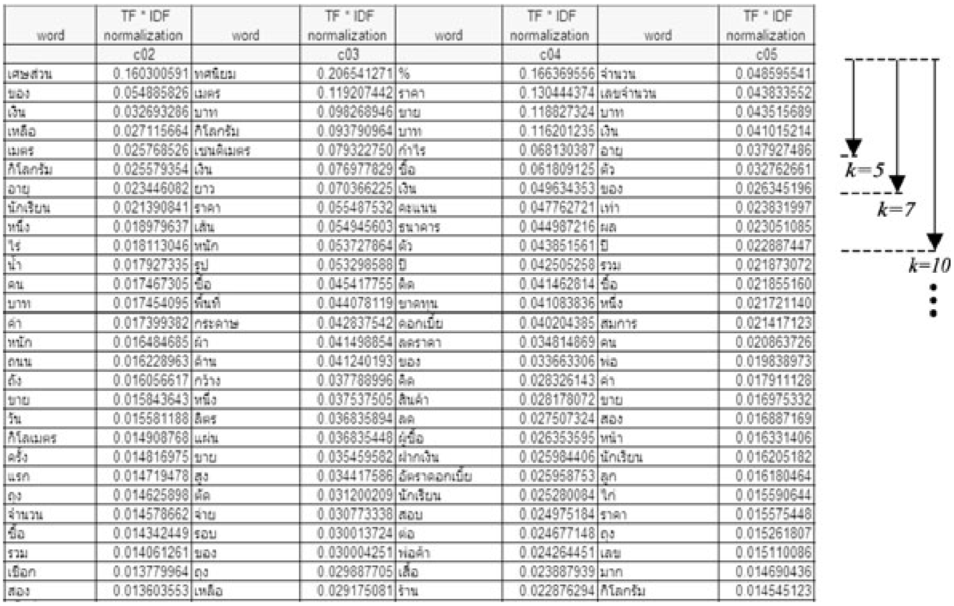
After the system obtains the weight of all terms in each category of the training set, the system ranks the weight in descending order by categories. All terms in the item bank still include irrelevant terms or noisy terms. These irrelevant or noisy terms may be the proper names or mistyping words. There are only a small percentage of the terms that are really meaningful for categorization. Some items can be categorized using only a highest weight term. To reduce the effect of noisy, we are supposed to extract the keywords from these terms. The keywords extraction not only reduces the effect of noisy that may decrease the categorization accuracy but also reduces the number of attributes of data and avoids overfitting. To select the attributes for each category, the two simple keyword selection strategies were implemented. Their results were compared. The first strategy was an attempt to answer to the question "How does the number of selected keywords affect the categorization result?" The second strategy is more naturally and similar to expert being. It is based on the concept that "Each category is not necessary to use the same number of selected keywords for categorization." The details of these two strategies are described below.

4.3.1. k-Highest Order Terms. The first strategy, we select the k-highest order terms according to their weights for each category. The value of k, selected keywords, affects the categorization result. The higher k value, the more keywords are extracted. If the value of k is too low that means we select only a few first-order terms, we may lose some actual keywords. On the other hand, if the value of k is too high we may get some noisy terms in the attributes. For both cases, the categorization performance can be affected. Figures 4 and 5 illustrate a selection algorithm and parts of the first strategy with some example categories, respectively.



```
Assign $k$ : = number of selected keyword (1 to 20)
For each $c_j$
    Rank all terms by $w_{j,k}$ in descending order
    Select $k$ terms from the top
End for
```

FIGURE 4. An algorithm of the first keyword selecting strategy.



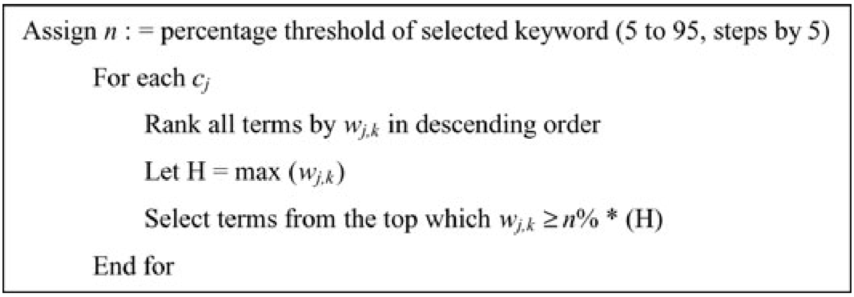
word	TF * IDF normalization	word	TF * IDF normalization	word	TF * IDF normalization	word	TF * IDF normalization
	c02		c03		c04		c05
เศษส่วน	0.160300591	ทศนิยม	0.206541271	%	0.166369556	จำนวน	0.048595541
ของ	0.054885826	เมตร	0.119207442	ราคา	0.130444374	เลขจำนวน	0.043833552
เงิน	0.032693286	บาท	0.098268946	ขาย	0.118827324	บาท	0.043515689
เหลือ	0.027115664	กิโลกรัม	0.093790964	บาท	0.116201235	เงิน	0.041015214
เมตร	0.025768526	เซนติเมตร	0.079322750	กำไร	0.068130387	อายุ	0.037927486
กิโลกรัม	0.025579354	เงิน	0.076977829	ซื้อ	0.061809125	ตัว	0.032762661
อายุ	0.023446082	ยาว	0.070366225	เงิน	0.049634353	ของ	0.026345196
นักเรียน	0.021390841	ราคา	0.055487532	คะแนน	0.047762721	เท่า	0.023831997
หนึ่ง	0.018979637	เส้น	0.054945603	ธนาคาร	0.044987216	ผล	0.023051085
ไร่	0.018113046	หนัก	0.053727864	ตัว	0.043851561	ปี	0.022887447
น้ำ	0.017927335	รูป	0.053298588	ปี	0.042505258	รวม	0.021873072
คน	0.017467305	ซื้อ	0.045417755	คิด	0.041462814	ซื้อ	0.021855160
บาท	0.017454095	พื้นที่	0.044078119	ขาดทุน	0.041083836	หนึ่ง	0.021721140
ค่า	0.017399382	กระดาษ	0.042837542	ดอกเบี้ย	0.040204385	สมการ	0.021417123
หนัก	0.016484685	ผ้า	0.041498854	ลดราคา	0.034814869	คน	0.020863726
ถนน	0.016228963	ด้าน	0.041240193	ของ	0.033663306	พ่อ	0.019838973
ถัง	0.016056617	กว้าง	0.037788996	คิด	0.028326143	ค่า	0.017911128
ขาย	0.015843643	หนึ่ง	0.037537505	สินค้า	0.028178072	ขาย	0.016975332
วัน	0.015581188	ลิตร	0.036835894	ลด	0.027507324	สอง	0.016887169
กิโลเมตร	0.014908768	แผ่น	0.036835448	ผู้ซื้อ	0.026353595	หน้า	0.016331406
ครั้ง	0.014816975	ขาย	0.035459582	ฝากเงิน	0.025984406	นักเรียน	0.016205182
แรก	0.014719478	สูง	0.034417586	อัตราดอกเบี้ย	0.025958753	ลูก	0.016180464
ถุง	0.014625898	ตัด	0.031200209	นักเรียน	0.025280084	ไก่	0.015590644
จำนวน	0.014578662	จ่าย	0.030773338	สอบ	0.024975184	ราคา	0.015575448
ซื้อ	0.014342449	รอบ	0.030013724	ต่อ	0.024677148	ถุง	0.015261807
รวม	0.014061261	ของ	0.030004251	พ่อค้า	0.024264451	เลข	0.015110086
เชือก	0.013779964	ถุง	0.029887705	เสื้อ	0.023887939	มาก	0.014690436
สอง	0.013603553	เหลือ	0.029175081	ร้าน	0.022876294	กิโลกรัม	0.014545123

FIGURE 5. Parts of the first keyword selecting strategy with some example categories.

4.3.2. n% of the Highest Term Weighting. In the second strategy, we select terms whose weights are equal to or greater than the percentage threshold (*n%* of the highest weights in each category) to be the keywords. The terms that have the lower weights are discarded. The lower *n* value, the more keywords are selected, and vice versa. If the value of *n* is too high we will lose too many relevant words that will decrease the categorization performance as well. Figure 6 shows a selection algorithm and Figure 7 illustrates parts of this strategy when $n = 30$ with some example categories.

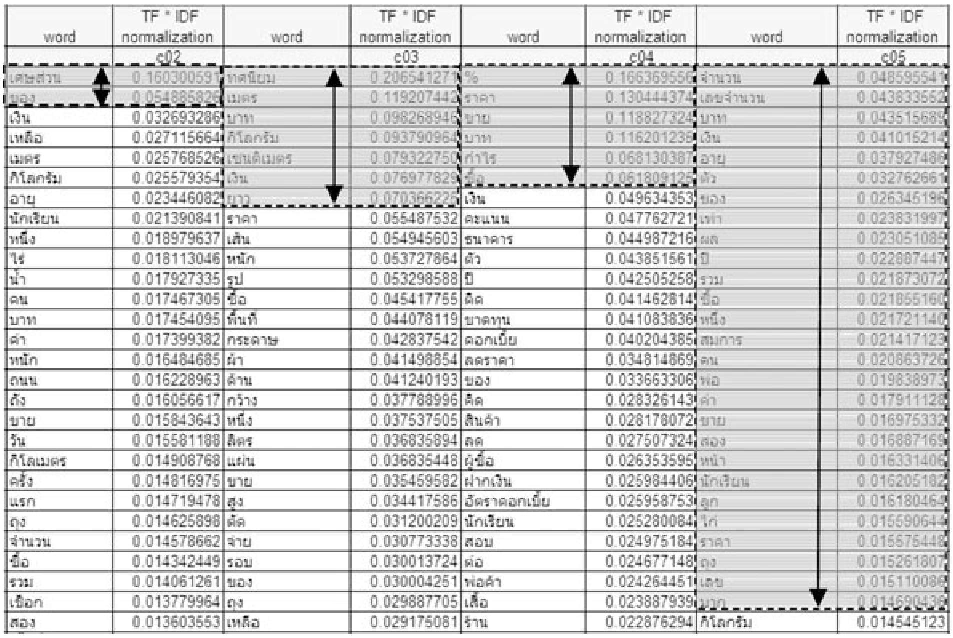
4.4. Patterned Keywords Generation

After all possible keywords are extracted from the items; they will be used to generate a keyword pattern for each item. In the patterned keywords generation process, any keyword



```
Assign $n$ : = percentage threshold of selected keyword (5 to 95, steps by 5)
    For each $c_j$
        Rank all terms by $w_{j,k}$ in descending order
        Let H = max ($w_{j,k}$)
        Select terms from the top which $w_{j,k} \geq n\% * (H)$
    End for
```

FIGURE 6. An algorithm of the second keyword selecting strategy.



word	TF * IDF normalization	word	TF * IDF normalization	word	TF * IDF normalization	word	TF * IDF normalization
	c02		c03		c04		c05
เศษส่วน	0.160300591	ทศนิยม	0.206541271	%	0.166369556	จำนวน	0.048595541
ของ	0.054885826	เมตร	0.119207442	ราคา	0.130444374	เลขจำนวน	0.043833562
เงิน	0.032693286	บาท	0.098268946	ขาย	0.118827324	บาท	0.043515689
เหลือ	0.027115664	กิโลกรัม	0.093790964	บาท	0.116201235	เงิน	0.041015214
เมตร	0.025768526	เซนติเมตร	0.079322750	กำไร	0.068130387	อายุ	0.037927486
กิโลกรัม	0.025579354	เงิน	0.076977829	ซื้อ	0.061809125	ตัว	0.032762661
อายุ	0.023446082	ยาว	0.070366225	เงิน	0.049634353	ของ	0.026345196
นักเรียน	0.021390841	ราคา	0.055487532	คะแนน	0.047762721	เท่า	0.023831997
หนึ่ง	0.018979637	เส้น	0.054945603	ธนาคาร	0.044987216	ผล	0.023051085
ไร่	0.018113046	หนัก	0.053727864	ตัว	0.043851561	ปี	0.022887447
น้ำ	0.017927335	รูป	0.053298588	ปี	0.042505258	รวม	0.021873072
คน	0.017467305	ซื้อ	0.045417755	คิด	0.041462814	ซื้อ	0.021855160
บาท	0.017454095	พื้นที่	0.044078119	ขาดทุน	0.041083836	หนึ่ง	0.021721140
ค่า	0.017399382	กระดาษ	0.042837542	ดอกเบี้ย	0.040204385	สมการ	0.021417123
หนัก	0.016484685	ผ้า	0.041498854	ลดราคา	0.034814869	คน	0.020863726
ถนน	0.016228963	ด้าน	0.041240193	ของ	0.033663306	พ่อ	0.019838973
ถัง	0.016056617	กว้าง	0.037788996	คิด	0.028326143	ค่า	0.017911128
ขาย	0.015843643	หนึ่ง	0.037537505	สินค้า	0.028178072	ขาย	0.016975332
วัน	0.015581188	ลิตร	0.036835894	ลด	0.027507324	สอง	0.016887169
กิโลเมตร	0.014908768	แผ่น	0.036835448	ผู้ซื้อ	0.026353595	หน้า	0.016331406
ครั้ง	0.014816975	ขาย	0.035459582	ฝากเงิน	0.025984406	นักเรียน	0.016205182
แรก	0.014719478	สูง	0.034417586	อัตราดอกเบี้ย	0.025958753	ลูก	0.016180464
ถุง	0.014625898	ตัด	0.031200209	นักเรียน	0.025280084	ไก่	0.015590644
จำนวน	0.014578662	จ่าย	0.030773338	สอบ	0.024975184	ราคา	0.015575448
ซื้อ	0.014342449	รอบ	0.030013724	ต่อ	0.024677148	ถุง	0.015261807
รวม	0.014061261	ของ	0.030004251	พ่อค้า	0.024264451	เลข	0.015110086
เชือก	0.013779964	ถุง	0.029887705	เสื้อ	0.023887939	มาก	0.014690436
สอง	0.013603553	เหลือ	0.029175081	ร้าน	0.022876294	กิโลกรัม	0.014545123

FIGURE 7. Parts of the second keyword selecting strategy when $n = 30$ with some example categories.

in each item (if any) is mapped to its order of appearance, in phrases of the item. This is why we called "PKIP, Patterned Keywords in Phrases." This keyword pattern is based on the basic concept that if two items have the same set of keywords and similar structure of keyword orders, they will have higher probability to be in the same category than others that have only the same keywords. For example, we describe three example items as follow.

Example 3:

Thai:
k_2 k_3 k_1
กระเบื้อง|แผ่น|หนึ่ง|มี|ด้าน|กว้าง|4|นิ้ว|ยาว|8|นิ้ว||จะ|มี|พื้นที่|เท่าไร
tile wide inch long inch area

English: What is the area of a tile that is 4 inches wide and 8 inches long?

PKIP: $\boxed{*/k_2/*/k_3/*/k_1/*}$ or */กว้าง/*/ยาว/*/พื้นที่/* (Thai)

Example 4:

Thai:
k_2 k_3 k_3 k_2 k_3 k_4
กระเบื้อง|แผ่น|หนึ่ง|กว้าง|10|ซม.||ด้าน|ยาว|ยาว|กว่า|ด้าน|กว้าง|20|ซม.||จะ|มี|ความ|ยาว|โดย|รอบ|เท่าไร
tile wide cm. long long wide cm. long around

English: How long is the border around a tile that is 10 cm wide and its length is 20 cm longer than the width?

PKIP: $\boxed{*/k_2/*/k_3/k_3/*/k_2/*/k_3/*/k_4/*}$ or */กว้าง/*/ยาว/ยาว/*/กว้าง/*/ยาว/*/รอบ/* (Thai)

Example 5:

Thai:
k_2 k_3 k_1
ต้อง|การ|ปู|พรม|บน|พื้น|ห้อง|ที่|กว้าง|4|ม.|ยาว|8|ม.||จง|หา|ว่า|จะ|ต้อง|ใช้|พรม|คิด|เป็น|พื้นที่|เท่าไร
carpet wide m. long m. carpet area

English: Find the area of a carpet that covers on the floor that is 4 meters wide and 8 meters long?

PKIP: $\boxed{*/k_2/*/k_3/*/k_1/*}$ or */กว้าง/*/ยาว/*/พื้นที่/* (Thai)

The items in Examples 3 and 5 were manually categorized into the same category, "area of quadrilateral," while an item in Example 4 was manually categorized into "perimeter of quadrilateral." Notice that an item in Example 3 shares more common words with an item in Example 4 than that in Example 5. Three keywords; k_1 ="พื้นที่" (area), k_2 ="กว้าง" (wide), k_3="ยาว" (long), were extracted from Example 3 and mapped to generate a keywords pattern as follows:

PKIP: $\boxed{*/k_2/*/k_3/*/k_1/*}$ or */กว้าง/*/ยาว/*/พื้นที่/*

Note: "$*$" represents any word(s).

Once PKIP for all three examples were generated, Examples 3 and 5 were assigned the same category because they shared the same PKIP, whereas Example 4 was assigned another category because its PKIP is different from those.

The number of PKIP for each category depends on the number of keywords and their order of appearance. In the case that a category contains *k* keywords, all possible number of pattern keywords is "*k*!" However, any ungrammatical patterns and any duplicate patterns will be deducted. Therefore, the total number of usable PKIPs is rather small and not greater than that of items in such category.

TABLE 1. Features in the Vector Space Model

Predefined Categories	Selected Keywords			Patterned Keywords		
	$\mathbf{k_1}$	$\mathbf{k_j}$	$\mathbf{k_n}$	$\mathbf{p_1}$	$\mathbf{p_k}$	$\mathbf{p_o}$
C_1	wk_{11}	wk_{1j}	wk_{1n}	b_{11}	b_{1k}	b_{1o}
...	...	...	...	...	...	...
C_i	wk_{i1}	wk_{ij}	wk_{in}	b_{i1}	b_{ik}	b_{io}
...	...	...	...	...	...	...
C_m	wk_{m1}	wk_{mj}	wk_{mn}	b_{m1}	b_{mk}	b_{mo}

4.5. Vector Space Model Generation

After completing the feature selection process, the weights of all keywords and the patterned keywords indicators (0, 1) of each category are combined and represented with the vector space model (VSM).

The features representation in the vector space model is designed as a feature matrix and is illustrated in Table 1.

In Table 1, C_i is a label of the predefined category i, there are m categories in total. k_i is an ith selected keyword, there are n keywords in total. wk_{mn} is a weight of the nth keyword in the mth category. p_i is an ith pattern, there are o patterns in total. The value of b_{mo} is either "0" or "1." It is "0" if p_o is not a pattern of category C_m and it is "1" if otherwise.

5. TEXT CATEGORIZATION

We adopted the SVM (Support Vector Machine) method as a classifier because this technique has proved significant improvement over other machine learning algorithms for text categorization (Joachims 1998). Furthermore, it gave top performance for text categorization (Yang and Liu 1999) and it gave the top precision in Thai text categorization as well (Murata, Ma, and Isahara 2002). The SVM categorization algorithm is a relatively new learning approach introduced by Cortes and Vapnik (1995) for solving two-class problems. It is based on the Structural Risk Minimization principle (Vapnik 1982). The SRM is one of the statistical learning theories. Its idea is to find a hypothesis space for which one can guarantee the lowest probability of error for a given training sample. The SVM method is defined over the vector space where the problem is to find a decision surface in hyper plane that best separates (that maximizes the margin) the data point into two classes. The SVM concept is illustrated in Figure 8 (Yang and Liu 1999). For simplicity, the figure shows a case in two-dimensional space. The solid lines show two possible decision surfaces that are correctly separates the two groups of data, black dots, and white dots. The distance between two parallel dashed lines is referred to the margin and the data points on the dashed lines are the Support Vectors. The aim of SVM is to find the decision surface that maximizes the margin.

Although SVM is an algorithm for solving two-class problems, a binary classifier, it can be applied to multiclass categorization tasks. In such case, for example m-classes problem, it is reduced to m binary tasks, which is called "one against the rest strategy." Another widely

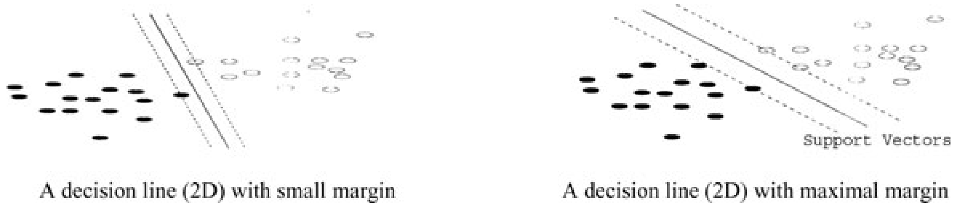

A decision line (2D) with small margin
Support Vectors
A decision line (2D) with maximal margin

FIGURE 8. SVM concept.

used strategy is "pair-wise" classification that reduces the problem into $m(m-1)/2$ binary tasks. The classifier is trained to discriminate the data between each pair of classes. The testing data are categorized by majority vote of all $m(m-1)/2$ predictions.

6. EVALUATION MEASUREMENT

To evaluate our approach, the SVM classifier model, which is called SMO function in Weka 3.5 (Witten and Frank 2005), an open source software collection of machine learning algorithms under the GNU License, was used as our classifier with linear function and no additional option.

With standard evaluation technique, stratified 10-fold cross-validation, the training data set was divided randomly into 10 parts. Each category was represented in approximately the same proportions as in the full training data set. The nine-tenth part was used to train the system while the remaining one-tenth part was used to test in turn. Thus procedure was executed 10 times on different training data sets. Then, the 10 evaluation results were averaged to estimate categorization results overall.

To evaluate the categorization results, we used the standard performance measure that is illustrated in terms of categorization accuracy, precision recall and F-measure. These measurements are defined by using a two-way contingency table as an example shown in Table 2.

Now we can define the values of accuracy, recall, and precision as follows:

$$Accuracy\ (Acc) = (a + d)/n \ where\ n = a + b + c + d > 0 \tag{8}$$

$$Recall(r) = a/(a + c)\ if\ a + c > 0, \quad otherwise\ undefined \tag{9}$$

TABLE 2. A Contingency Table

Categorization Result	YES Is Correct	NO Is Correct
Assigned YES	a	b
Assigned NO	c	d

where
a = number of the items correctly assigned by the system to this category.
b = number of the items incorrectly assigned by the system to this category.
c = number of the items incorrectly rejected by the system from this category.
d = number of the items correctly rejected by the system from this category.

$$Precision\ (p) = a/(a+b)\ if\ a+b > 0, \quad otherwise\ undefined \tag{10}$$

Moreover, $F_{1,}$ a statistical average value, is used to evaluate the combined measure that depends on both recall and precision values.

$$F - measure\ (F_1) = \frac{2 * p * r}{p + r} \tag{11}$$

7. EXPERIMENTS AND RESULTS

In our initial experiment, we applied PKIP with the moderate Thai primary mathematics problems collected from many published mathematics practiced books and some existing mathematics item banks. Our training data set contains 4892 items in 31 different categories. The number of items in each category varies widely, 791 items in the largest category and five items in the smallest category (Table 3). Many items are assigned into more than one category. The statistical mean number of categories assigned to an item is 1.5. This training data set was evaluated by stratified 10-fold cross-validation.

The experiment began with selecting the features of the training data set, categorizing it by SVM and evaluating the categorized items.

Because the question items in this experiment are in Thai language, the three preprocessing including word segmentation, substitution function and stop word elimination were applied. The stop words list was developed based on the research of Jaruskulchai (1998). The weights of all words were calculated and sorted in descending order for each category.

The two keywords extraction strategies were applied separately. For the first strategy, we selected k highest-order words from each category with $k = 1$ to 20. Figure 9 presents that when the number of keywords increases, the number of their patterns occurred in every item increases also in logarithmic function and thus causes the number of unmatched items

TABLE 3. Number of Training Items of Five Largest and Five Smallest Categories

Category Number and Name (Five Largest Categories)	Number of Items
C03 Decimal number	791
C00 Number and basic calculation	612
C04 Rule of three application, Percentage	427
C01 Factor, H.C.F., L.C.M.	367
C12 Area of quadrilateral	334
Category Number and Name (Five Smallest Categories)	**Number of Items**
C31 General topics of Quadrilateral	15
C30 General topics of Triangle	12
C25 Circular cone	10
C32 General topics of Circle	8
C26 Sphere	5

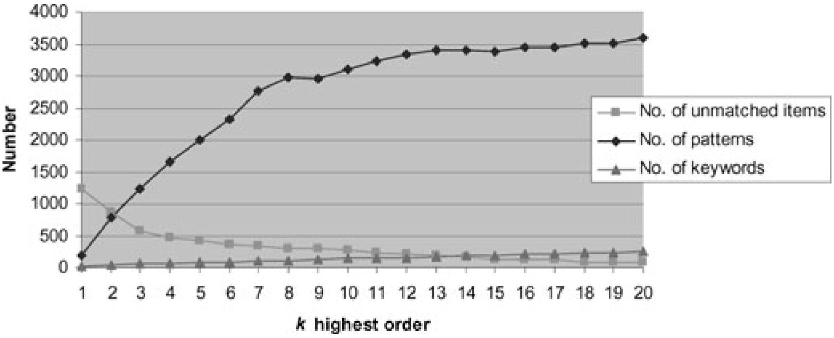


FIGURE 9. Number of unmatched items, keywords, and their patterns with various k highest orders in data set.

TABLE 4. The Categorization Result with Various k Highest Orders

k-Highest Order	Correctly Categorized Items (%)	Macro Average Precision (%)	Macro Average Recall (%)	Macro Average F_1 (%)	Std. Dev. (σ) of F_1 of All Categories (%)
1	92.14	88.32	89.84	88.35	±28.57
2	94.34	91.11	90.09	90.37	±19.89
3	90.59	91.90	88.89	89.74	±14.02
4	85.21	89.69	86.11	87.25	±22.16
5	85.05	87.05	83.75	85.03	±22.76
6	84.95	87.79	85.02	86.06	±22.42
7	92.80	95.14	90.97	92.67	±08.51
8	93.83	95.14	92.94	93.90	±06.66
9	85.16	89.34	86.95	87.98	±20.07
10	87.61	91.75	88.95	90.14	±14.79
11	87.76	90.74	87.69	88.95	±15.71
12	88.29	89.95	87.17	88.32	±14.07
13	87.53	91.17	87.89	89.27	±11.38
14	86.19	90.48	87.27	88.63	±13.90
15	83.10	87.88	84.55	85.99	±20.24
16	84.62	89.30	85.69	87.24	±16.41
17	82.90	87.25	83.12	84.92	±17.55
18	82.62	87.15	83.25	84.87	±15.59
19	81.43	85.91	81.84	83.52	±17.28
20	82.30	86.46	82.95	84.48	±15.42

decreases. Note that the sum of number of keywords and number of their patterns are the number of attributes or the size of our data set vector.

The results in Table 4 show that the percentage of correctly categorized items, precision, recall and F_1 fluctuate with the k value. The four highest correctly categorized items are 94.34%, 93.83%, 92.80%, and 92.14% when $k = 2$, 8, 7, and 1, respectively. However, it seems that the best performance is at $k = 8$ because all measurement values are very high and the standard deviation is the lowest.

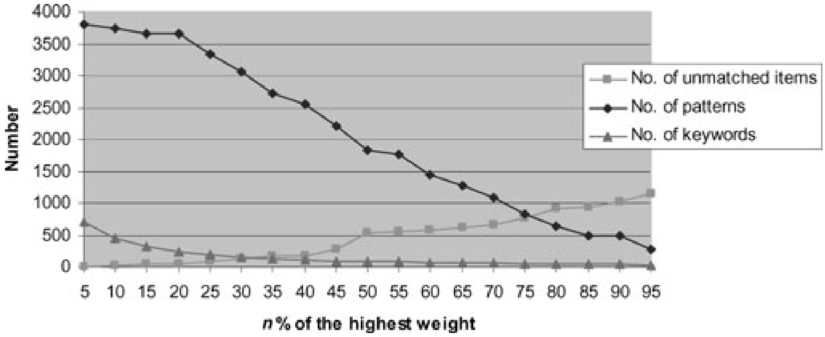


FIGURE 10. Number of unmatched items, keywords, and their patterns with various *n*% in data set.

In applying the second keywords extraction strategy, we select keywords from each category by their weights that equal to or greater than *n*% of the highest weight in their categories. From the observation of our data set, we use $n = 5$ to 95, step by 5. Although *n* increases, the number of unmatched items increases but the numbers of all keywords and their patterns decreases as shown in Figure 10.

The results in Table 5 show that the percentage of correctly categorized items, precision, recall, and F_1 fluctuate with the *n* value. The five highest correctly categorized items are given in descending order 94.56%, 93.47%, 91.93%, 90.88%, and 90.39% when $n = 55$,

TABLE 5. The Categorization Result with Various *n* %

n	Correctly Categorized Items (%)	Macro Average Precision (%)	Macro Average Recall (%)	Macro Average F_1 (%)	Std. Dev. (σ) of F_1 of All Categories (%)
5	63.24	55.79	51.09	52.80	±28.10
10	73.76	73.04	68.90	70.60	±22.00
15	81.20	84.10	80.70	82.10	±18.80
20	89.09	91.90	89.06	90.20	±09.70
25	84.69	89.78	86.81	87.90	±19.80
30	88.79	92.91	89.18	90.20	±17.20
35	88.47	91.87	89.30	90.30	±21.40
40	93.47	95.14	91.93	93.20	±10.80
45	88.89	92.07	89.57	90.70	±17.30
50	87.08	91.61	88.52	89.20	±22.80
55	94.56	91.63	91.27	91.30	±18.60
60	87.34	87.97	87.29	87.50	±26.20
65	87.20	87.61	86.99	87.10	±25.60
70	89.65	88.27	87.19	87.00	±24.50
75	86.48	89.63	88.82	88.40	±22.90
80	85.88	87.52	87.16	86.40	±26.10
85	90.88	91.11	90.47	89.90	±21.70
90	91.93	91.11	91.28	90.70	±21.30
95	90.39	86.14	88.80	86.90	±28.90

TABLE 6. Categorization Result of the Five Largest and Five Smallest Items Categories

	The First Keywords Extraction Strategy, When $k = 8$				The Second Keywords Extraction Strategy, When $n = 40$			
Category Number and Name	Accuracy (%)	Precision (%)	Recall (%)	F_1 (%)	Accuracy (%)	Precision (%)	Recall (%)	F_1 (%)
C03 decimal number	99.94	99.70	99.90	99.80	100.00	100.00	100.00	100.00
C00 number and basic calculation	97.19	81.70	93.40	87.10	97.37	84.50	91.40	87.80
C04 rule of three application and percentage	99.96	100.00	99.50	99.80	99.87	99.00	99.50	99.30
C01 factor & H.C.F. & L.C.M.	98.91	93.80	88.60	91.10	99.58	98.80	95.50	97.10
C12 area of quadrilateral	99.30	92.10	98.20	95.10	99.66	96.20	97.30	96.70
C31 general topics of quadrilateral	99.98	100.00	93.30	96.60	99.98	100.00	93.30	96.60
C30 general topics of triangle	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00
C25 circular cone	99.96	90.00	90.00	90.00	99.96	100.00	80.00	88.90
C32 general topics of circle	99.94	85.70	75.00	80.00	99.87	75.00	37.50	50.00
C26 sphere	99.98	100.00	80.00	88.90	100.00	100.00	100.00	100.00

40, 90, 85, and 95, respectively. All high performance values are at $n = 40$ with the lowest standard deviation.

Table 6 shows the categorization performance of the five largest and five smallest items categories, comparing between the two keywords extraction strategies. The categories "Number and basic calculation" and "Factor & H.C.F. & L.C.M." of the five largest groups and the categories "Circular cone" and "General topics of Circle" of the five smallest groups presents lower categorization performance because the contents of some items in these categories are overlap. Consequently, they share some similar features that are confusing the classifier. For this reason, their categorization performance is dropped. The overall categorization results show that both keywords extraction strategies provide about the same categorization performance when using with PKIP approach. Although the first strategy is slightly better than the second strategy in correctly categorizing the items, it discarded 306 unmatched items and had 3099 attributes for data set vector size whereas the second strategy discarded only 176 unmatched items and had 2663 attributes for vector size.

8. CONCLUSION

An alternative approach, PKIP, a feature-based selection was proposed. PKIP was designed to categorize a question item using only available keywords and their patterns. The feature selection adopted the term weighting function, *tf∗idf*, of all items in the same category to extract the keywords. PKIP identifies the appropriate keywords by two strategies: k-highest order terms and $n\%$ of the highest term weighing.

Different selection strategies cause the different number of unmatched items and number of attributes or vector size. With stratified 10-fold cross-validation, the categorization results show that the both keywords extraction strategies, with PKIP, give high categorization performance. The best categorization results of the two keywords selection strategies are obtained when k is 8 and n is 40%. In this experiment, "n% of the highest term weighting" provided approximately 14% smaller vector size and discarded approximately 42% less number of unmatched items than "k-highest order terms" did. With fine-tuning the parameters k and n, the proposed keywords extraction strategies can improve overall categorization performance for any particular item banks.

REFERENCES

CORTES, C., and V. VAPNIK. 1995. Support vector machines. Machine Learning, **20**:273–297.

GHANEM, M. M., Y. GUO, H. LODHI, and Y. ZHANG. 2002. Automatic scientific text classification using local patterns: KDD CUP 2002 (Task 1), SIGKDD Explorations. ACM SIGKDD, **4**(2):95–96.

GIORGETTI, D., and F. SEBASTIANI. 2003. Automating survey coding by multiclass text categorization techniques. Journal of the American Society for Information Science and Technology, **54**(14):1269–1277.

JARUSKULCHAI, C. 1998. An automatic indexing for thai text retrieval, Ph. D. Thesis, The School of Engineering and Applied Science, The George Washington University, Washington, DC, 59–60.

JING, L., H. HUANG, and H. SHI. 2002. Improved feature selection approach TFIDF in text mining. *In* Proceedings of the 1st international conference on machine learning and cybernetics, Beijing, China.

JOACHIMS, T. 1997. A probabilistic analysis of the rocchio algorithm with TFIDF for text categorization. *In* Proceedings of the 14th international conference on machine learning, pp. 143–151, Nashville, TN.

JOACHIMS, T. 1998. Text Categorization with support vector machines: Learning with many relevant features. *In* Proceedings of ECML-98, pp. 137–142, Chemnitz, Germany.

JOACHIMS, T. 2002. Learning to classify text using support vector machines. Kluwer Academic Publishers, Boston, pp. 12–16.

LEWIS, D. 1992. An evaluation of phrasal and clustered representations on a text categorization task. *In* Proceedings of the 15th annual international ACM SIGIR conference on research and development in information retrieval, pp. 37–50, Copenhagen, Denmark.

LI, X., and D. ROTH. 2002. Learning question classifiers. *In* Proceedings of the 19th international conference on computational linguistics (COLING), pp. 556–562, Taipei, Taiwan.

MURATA, M., Q. MA, and H. ISAHARA. 2002. Comparison of three machine-learning methods for Thai part-of-speech tagging. ACM Transactions on Asian Language Information Processing, **1**(2):145–158.

SAKURAI, S., and A. SUYAMA. 2004. Rule discovery from textual data based on key phase pattern, ACM Symposium on Applied Computing, SAC'04, March 14–17, pp. 606–612, Nicosia, Cyprus.

SALTON, G., and C. BUCKLEY. 1988. Term-weighting approaches in automatic text retrieval. Information Processing and Management, **24**(5):513–523.

VAPNIK, V. 1982. Estimation of dependencies based on empirical data. Springer-Verlag, New York.

WIGGINS, G. 1998. Educative assessment designing assessments to inform and improve student performance. Jossey-Bass, San Francisco.

WITTEN, I. H., and E. FRANK. 2005. Data Mining: Practical Machine Learning Tools and Techniques (2nd ed.). Morgan Kaufmann, San Francisco.

YANG, Y., and X. LIU. 1999. A re-examination of text categorization methods. *In* Proceedings of the 22nd international conference on research and development in information retrieval, pp. 42–49, Berkeley, CA.

ZHANG, D., and W. SUN LEE. 2003. Question classification using support vector machines. *In* Proceedings of the 26th annual international ACM SIGIR conference on research and development in information retrieval, pp. 26–32, Toronto, Canada.