KIP: A Keyphrase Identification Program with Learning Functions

Yi-fang Brook Wu, Quanzhi Li, Razvan Stefan Bot, Xin Chen

Information Systems Department
New Jersey Institute of Technology
{wu, ql23, rsb2, xc7}@njit.edu

Abstract

In this paper, we report a keyphrase identification program (KIP) which uses sample human keyphrases and then learns to identify additional new keyphrases. KIP first populates its database using manually identified keyphrases; each keyphrase is pre-processed and assigned an initial weight. It then extracts noun phrases from documents. All noun phrases will be assigned a score, depending on the weights for words it contains; the ones that have a score higher than the threshold will be selected as keyphrases. Learned new keyphrases will be inserted to the database and weights will be updated. As a result, new keyphrase identification iteration will be triggered. The process stops when no new keyphrases are identified during previous iteration. According to the results of evaluation, the base KIP system’s average recall was 0.7 and precision was 0.44. The augmented KIP with learning functions did produce new keyphrases which were not identified by the base system.

1. Introduction

With the amount of textual information available on the web and in corporate databases, the importance of text mining to deriving competitive advantages cannot be ignored. However, the challenge of text mining is the difficulty of finding important elements in documents for analysis. Most commercial text mining applications do not have an adequate natural language processing function beyond simple automatic keyword indexing. The consequence is that there are too many textual elements, basically single words other than stopped words, without any context attached; it also makes deriving any meaningful associations between them difficult. Instead of using single words as textual elements, we argue that noun phrases (NPs) are better suited for text mining. Evidences from language learning of children [10] and discourse analysis theories (e.g. Discourse Representation Theory) show that the primary concepts in text are carried by noun phrases [7]. Noun phrases are considered the conceptual entities in text messages. However, not all NPs are important in a document. We propose to identify keyphrases from a list of NPs obtained from a document.

All keyphrase identification studies use some forms of human judgments as the foundation for keyphrase selection. Some require selecting initial keyphrases from documents, e.g. authors’ keywords; some use controlled vocabularies, e.g. thesauri, or glossaries, to identify and assign keyphrases. The latter will easily be a problem because most controlled vocabularies are not updated frequently enough; in some domains, such controlled vocabularies might not even be available. It becomes imperative for a keyphrase identification technique to adapt to the new development and advances of the domain of documents it tries to derive keyphrases from.

In this paper, we discuss a domain-specific keyphrase identification program (KIP) with learning functions. It takes human judgments in any form, e.g. authors’ keywords or thesauri, to populate the keyphrase database and assign initial weights. Once KIP identifies new keyphrases from documents, the weights for keyphrases are updated and iteration will be repeated to identify new keyphrases using the updated weights. The learning and identification process stops when no new keyphrases are found. In section 2, we discuss prior studies. Section 3 illustrates our methodology and system architecture. Evaluation and results are reported in Section 4. Section 5 gives a brief conclusion.

2. Related work
Kruplutch and Burkey [8] extract significant phrases from a document using heuristics. The heuristics are based on documents’ structural features, such as the use of italics. Their motivation is to use the extracted phrases in automatic document classification.

Turney [11, 12] is the first to treat the problem of phrase extraction as supervised learning from examples. Keyphrases are extracted from candidate phrases based on examination of their features. Turney introduces two kinds of algorithms: the first of which is C4.5 decision tree induction algorithm, and the second one is GenEx, which consists of two components: Extractor and Gentor. GenEx is more successful than C4.5. Extractor processes a document and produces a list of phrases based on the setting of 12 parameters. In the training stage, Gentor is used to tune the parameter setting of the 12 parameters to get the optimal performance. Once the training process is finished, Gentor is no longer used and Extractor alone can extract keyphrases using the optimal parameter setting obtained from training stage.

Kea is a keyphrase extraction program using another machine learning algorithm based on naïve Bayes’ decision rule [4]. Kea uses a model to identify the keyphrases within a document. The model is learned from the training documents with exemplar keyphrases. Two attributes are used to identify keyphrases: the distance into a document that a phrase first occurs, and its TD×IDF value.

Barker and Cornacchia’s [1] B&C keyphrase extraction system consists of three steps: skimming a document to find the base noun phrases, assigning scores to noun phrase based on their length and frequency, and filtering some noise from the top scoring keyphrases. A noun phrase’s score is the product of its length (number of words) multiplied by its frequency. In the final step, the system applies two simple post-processing filters to the top keyphrase candidates: remove single letter keyphrases and sub phrases which are wholly contained within longer phrases.

3. The KIP algorithm

Our design of the KIP algorithm considers the composition of a keyphrase. If a noun phrase contains a keyword, it is likely that the noun phrase is a keyphrase candidate. The more keywords it contains, the more likely it is a keyphrase. In order to use the composition of noun phrases to identify keyphrases, we need to parse readily available human identified keyphrases to form a keyword database for weight calculation. Our design also considers the learning function, so that it adapts to the advances of a domain, meaning the keyphrase database grows as more documents are processed. KIP has the following main processes: part of speech tagging, noun phrase extraction, keyphrase extraction, and learning.

3.1. Part of speech tagging

In this stage, documents are loaded into the system and tokenized to obtain the atom units. Then, each word is assigned with an initial part-of-speech (POS) tag. To assign the right tag we use a simplified WordNet lexical database, which contains words divided into four categories (noun, verb, adjective, and adverb) and the number of senses of each word used in the categories it belongs to. If a word is found in more than one category, it is marked as a multi-tag word. The initial POS tag for a word is determined by the category having the maximum number of senses of this word. The next step is multi-tag disambiguation. For every multi-tag word, the sequence of the POS tags of the proceeding n tokens (n ranges from 2 to 4) is examined against a list of predefined syntactic rules. If a word is not found in any of the categories and its POS tag cannot be solved by the syntactic rules, some heuristics are used to determine its POS tag.

3.2. Noun phrase extraction

In general, a noun phrase means a sequence of words that usually gives us very useful information. People mostly use noun phrase as concept terms. After tagging the text, the noun phrase extractor extracts noun phrases by selecting the sequence of POS tags that are of interests. The current sequence pattern is defined as [A] {N}, where A refers to Adjective, N refers to Noun, [ ] means optional, and { } means repetition. A set of exceptional rules is used as well. The KIP system has a system parameter to set the minimum number of words of a noun phrase. By changing the parameter value user can get noun phrases with different length.

3.3. Keyphrase extraction

At the keyphrase extraction stage, all the identified noun phrases are scored and ranked. The number of extracted keyphrases is based on the system settings and/or user’s preference. To assign scores to noun phrases, the system needs to access a database that stores the domain-specific keyphrases and keywords.
In the following sections, we describe how the database is built, how a noun phrase is assigned a score, and how the keyphrases are extracted.

3.3.1. Domain-specific keyphrase and keyword database. This database contains two lists (tables): a keyphrase list and a keyword list. Keyphrase means a domain-specific phrase that consists of one or more words identified by human experts (from authors’ keywords, a thesaurus, or a glossary), and a keyword means a single domain-specific word parsed from keyphrase list. Different domains will need different keyphrase and keyword databases. When the system is applied to a new domain the only thing to do is to build or change to a new database specific to this domain. Each table has two columns. The first column represents the keyphrase or keyword, and the second column represents the assigned weight for a keyphrase or keyword. The weights for keyphrases and keywords are assigned automatically. The rationale behind the method of assigning weights is that it reflects how specific a keyword or keyphrase is in a specific domain.

KIP will use the weights of keyphrases and keywords in the database to calculate the scores of noun phrases in a document.

3.3.2. Calculating noun phrase score. At this stage, the system already has a list of noun phrases for a document. The calculation process is described below.

A noun phrase’s score is defined as the sum of weights of all the words and all the possible combinations of adjacent words within the noun phrase. The following example is used to explain how a noun phrase’s score is calculated. Assume there is a noun phrase “ABC,” where A, B and C are three words. The possible combinations of adjacent words are AB, BC and ABC. The score for noun phrase “ABC” will be the summation of weights of A, B, C, AB, BC and ABC.

To obtain the weight for each word, the system will lookup the keyword table. If it finds the word in the table, the weight of this word will be the corresponding weight in the database. Otherwise, a very low predefined weight will be assigned to this word. To obtain the weight for the combination of adjacent words, which is also a sub phrase of the noun phrase, the system will lookup the keyphrase table. If the sub phrase is found in the table, the corresponding weight in the table is assigned to this sub phrase; otherwise, a predefined weight will be assigned to this sub phrase.

3.3.3. Extracting keyphrases. After the scores of all noun phrases in a document are calculated, they are normalized and their values range from 0 to 1. All the noun phrases are ranked in a descending order according to their scores. The keyphrases of a document can be extracted from the ranked noun phrase list. Depending on the circumstance under which the keyphrase extraction program is used, the number of extracted keyphrases will be different. It also depends on the characteristics of a document, such as its length. In order to be as flexible as possible on the number of keyphrases extracted, the KIP system has a set of parameter settings to let the users decide how many keyphrases they want or leave it to the system to produce a optimal number of keyphrases for a specific document.

3.4 Adaptation and Learning

Many keyphrase extraction research efforts rely on human identified keyphrases as positive examples. This could be a problem when such examples are not available or not up to date. It is, therefore, very important for KIP to have adaptation and learning functions, so it grows as does the field of documents.

Figure 1. Screenshot of KIP’s learning function
This function is used to enrich the database by automatically adding new keyphrases and keywords to the database. It affects the output of current document, as well as the outputs of future keyphrase extraction process. Whenever the system identifies a new keyphrase, which means this keyphrase as a single unit is not found in database’s keyphrase table, and it satisfies the inclusion requirements, this keyphrase will be added to the database as a keyphrase. The system will recalculate and reassign weights to the affected keyphrases and keywords in the database.

With this feature the database grows gradually and the system performance will be improved. This will benefit future keyphrase extraction on new documents. This function is especially useful when KIP is used in a domain where there is very few existing domain-specific keyphrases and keywords. In this situation, the KIP can automatically learn new phrases and keywords specific to this domain, and finally the KIP can build a thesaurus for this domain.

Figure 1 shows a screenshot for the system’s learning feature. The system processes a document and six keyphrases are extracted in the first iteration. The system finds that all six phrases are new to the database and adds them into the database. The learning operation is an iterative process. After the system adds the new phrases into database, it will extract keyphrases again for the current document using the updated database.

4. Evaluation

There are two basic approaches to evaluate automatically extracted keyphrases [6]. The first one uses the standard information retrieval measures: precision and recall. The second one involves human evaluation of extracted keyphrases. We used both of them.

KIP was evaluated using a collection of abstracts from Communications of the ACM, Information Systems Research and Journal of Management Information Systems. Documents used in the evaluation were preprocessed by keeping only the title and the abstract. 150 documents were selected from the collection and preprocessed. 75 of them were used in evaluation 1; other 75 were used in evaluation 2.

4.1. Evaluation 1

We recruited 3 fourth-year PhD students whose major is information systems to be the domain experts. Each of them were given 25 documents and asked to choose keyphrases from the document that best represent this document.

We used phrases manually identified by domain experts, instead of author keywords, as the pre-defined keyphrases to calculate the precision and recall. The number of automatically extracted keyphrases by KIP was 10 for this evaluation. The results of precision and recall are presented in Table 1.

<table>
<thead>
<tr>
<th>Table 1. Average precision and recall</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean (75 documents)</td>
</tr>
<tr>
<td>--------------------------------------</td>
</tr>
<tr>
<td>Precision: 0.44</td>
</tr>
<tr>
<td>Recall: 0.70</td>
</tr>
<tr>
<td>Standard Deviation</td>
</tr>
<tr>
<td>Precision: 0.17</td>
</tr>
<tr>
<td>Recall: 0.18</td>
</tr>
</tbody>
</table>

Most of the previous study use author-provided phrases to measure precision and recall. Table 2 presents the results of some previous studies and KIP [6, 11].

<table>
<thead>
<tr>
<th>Table 2. Precisions and recalls</th>
</tr>
</thead>
<tbody>
<tr>
<td>System</td>
</tr>
<tr>
<td>Kea</td>
</tr>
<tr>
<td>Extractor</td>
</tr>
<tr>
<td>0.14</td>
</tr>
<tr>
<td>KIP</td>
</tr>
<tr>
<td>0.13</td>
</tr>
</tbody>
</table>

4.2. Evaluation 2

We used the same 3 PhD students described in evaluation 1 as experts. Each subject was given all of the 75 documents and the corresponding extracted keyphrase lists. The experts were required first to read a document fully and then go over the extracted keyphrase list for the document. For each keyphrase, the subject rated the quality of the keyphrase, in term of how relevant to the document it is, using a five-point scale that ranges from 1 (worst) to 5 (best).

We calculated the mean score of the keyphrases for each document and also the mean score for all the documents when the number of extracted keyphrases for each document was 2, 4, 6, 8, and 10 respectively. The acceptable rate of the extracted was also calculated.

Generally, in experiment involving human judgments, there is analysis on subject agreement. We used two approaches to measure the inter-judge agreement: the Kappa Statistic K [2] and the Kendall Coefficient of Concordance W [9]. The value of K is
between 0 and 1. A value of 1 reflects complete agreement between the subjects and a value of 0 indicates a chance agreement. The average K value for all the 75 documents is 0.30. A drawback of the K method is that it considers agreement on unordered categories [6]. To avoid this problem, we also used the Kendall Coefficient of Concordance W method, which could measure the agreement between subject’s relative rankings of keyphrases. The average W value for all the 75 documents is 0.85.

Table 3 shows the average scores for the 75 documents when the number of extracted keyphrase is 2, 4, 6, 8 and 10.

### Table 3. The mean scores of KIP.

<table>
<thead>
<tr>
<th>Number of extracted keyphrases</th>
<th>Mean score</th>
<th>Standard deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>4.12</td>
<td>0.35</td>
</tr>
<tr>
<td>8</td>
<td>4.17</td>
<td>0.36</td>
</tr>
<tr>
<td>6</td>
<td>4.31</td>
<td>0.38</td>
</tr>
<tr>
<td>4</td>
<td>4.35</td>
<td>0.41</td>
</tr>
<tr>
<td>2</td>
<td>4.51</td>
<td>0.46</td>
</tr>
</tbody>
</table>

Turney conducted an evaluation of Extractor on the Web using 205 self-selecting human assessors [11]. In their study, subjects were asked to evaluate the extracted keyphrases as “good,” “bad” or “no opinion.” The number of extracted keyphrases was fixed at 7. Turney claimed that about 82% of the extracted keyphrases were acceptable (acceptable meaning not bad). Jones and Paynter [5] also did a similar evaluation of Kea. When we restricted the number of generated keyphrases of the KIP to also 7, According to the Turney’s definition of acceptable, on average 94% (80.3% + 13.7%) of the keyphrases extracted by KIP is acceptable. Table 4 shows the results of these three systems.

### Table 4. Comparison of acceptable rates

<table>
<thead>
<tr>
<th>System</th>
<th>Good (%)</th>
<th>Bad (%)</th>
<th>No opinion (%)</th>
<th>Acceptable (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Extractor</td>
<td>62</td>
<td>18.1</td>
<td>19.9</td>
<td>81.9</td>
</tr>
<tr>
<td>Kea</td>
<td>64.6</td>
<td>27.5</td>
<td>7.9</td>
<td>72.5</td>
</tr>
<tr>
<td>KIP</td>
<td>80.3</td>
<td>6</td>
<td>13.7</td>
<td>94</td>
</tr>
</tbody>
</table>

5. Conclusion

In this paper, we have introduced and evaluated a new algorithm, KIP, for automatically extracting keyphrase from documents. This algorithm is based on domain-specific keyphrases and keywords. Using the evaluation results, we also show that KIP can outperform other state-of-the-art keyphrase extraction algorithms. Human evaluation results of the keyphrases generated by KIP suggest that about 94% of the keyphrases are acceptable. The ranking of scores produced by KIP is effective. KIP’s ability to automatically learn domain-specific phrases and keywords will make it easier to apply KIP to different domains. The features and performance of KIP will make it useful for a variety of applications, such as text mining and subject metadata selection for documents.

6. References
