Using Mutual Information to Identify New Features for Text documents of Various Domains

GUO Zhi Li
IBM China Research Lab
2/F, Haohai Building, No. 7, Shangdi 5th Street, Haidian District
Beijing 100085, China
guozhili@cn.ibm.com

Abstract

The task of identifying proper names, unknown words and new terms, is an important step in text processing systems. This paper describes a method of using mutual information to collect possible segments as candidates of these three feature types in a document scope. Then the construction and context of each possible feature is examined to determine its type, canonical form and meaning. Adding very little domain-specific knowledge, this method adapts to various domains easily.

1 Introduction

Most text processing systems, e.g., information extraction, text categorization, clustering, and machine translation, use words (rather than characters) as basic units to build their algorithms. Thus morphological process like segmentation and feature identification becomes an important step, extremely for languages like Chinese, which lacks morphological marks like space separator between words or capital letters at proper names. The quality of segmentation and feature identification greatly influences the performance of the overall text processing systems.

Unknown words, i.e., words used in a document but not collected in a segmentation dictionary, and unknown proper names (persons, locations, organizations and their abbreviations) often reduce the precision and quality of a Chinese segmentation algorithm. Although there are ways to collect a balanced word list as the basis of a dictionary, a document always has new words and new names. Hence there must be an efficient way to identify these features. If they are not correctly identified, the real single-character words and the lone characters that actually combine into a new word, the common words and proper names will all be mixed together, which impediments followed processing steps.

This paper presents a method of using document-scope mutual information to identify three types of features in a unified algorithm. These three types of features are:

- proper names, including sub-categories of person names, place names, organization names, and their abbreviation forms;
- unknown words, including common words used in a document but not collected in segmentation dictionary, product and brand names;
- document terms, usually some phrases formed by multiple words. These are often key concepts of a document.

Mutual information is used to evaluate the coherence level of two consecutive characters and words in a document, and those bigrams of higher mutual information are assumed as "seed" of possible features. Then these seeds are extended to both their right and left sides, still using mutual information as a criterion to determine how long to be extended, to form a list of candidate features. The sifted patterns are assigned a category type and a confidence value, according to their internal constructions, their contexts, and their distribution in the document scope.

Since the method uses document-scope statistics as a major criterion to identify proper names, unknown words, and document terms, it's relatively easy to adjust to various domains, by adding

relatively little domain-specific knowledge represented via rules. The document statistics results can also be a helpful resource to assist generating these rules. The method proposed in this paper doesn't require a large manually-tagged corpus, but it could cooperate with such a corpus, in a way of learning useful domain-knowledge from corpus and then applying it to this method.

The next section introduces some previous works on feature identification, focusing on that for Chinese language. Section 3 introduces the method to calculate entropy and mutual information values for arbitrary length of patterns. Section 4 introduces the overall steps of feature identification. Section 5 provides the results of some preliminary experiments, and the last section gives some thoughts about future works.

2 Background and Previous Work

Most previous efforts of feature identification fall into two categories. The first category, which has been studied for a long time, is mainly knowledge based, i.e., they utilize various language knowledge (often represented in expert-collected rules) to identify a single type of feature. The second category is based on some kind of machine learning or corpus statistics method, aiming at identifying all manually-tagged feature types in the corpus.

Sun (1995) describes a method to identify Chinese person names based on several character lists, e.g., using a list of surnames, two lists of commonly used given-name characters, and a list of commonly used given-name words. Ji (2001) combined a probalistic model called inverse name frequency model and language rules to identify Chinese person names. Tan(1999) proposed a knowledge-based Chinese place name identification method. Zhang(1997) described a rule-based method to identify Chinese university and college names, using a list of suffix words and some pattern rules.

The method of using machine learning to identify language features is relatively newer. Baluja(1999) describes a method using 29 language cues to identify English proper names. The cues include word marks, dictionary lookup result, part-of-speech, and a word's adjacency to punctuation marks; but they are apparently chosen for English language, such as whether a word is upper-cased and whether a token is found in the dictionary. This kind of method requires much less language expertise, but it needs a sufficiently large mature corpus. Also the training result is usually hard for human beings to understand, thus it's relatively hard to integrate expert knowledge with the machine learning method.

In contrast to the above two categories of feature identification method, this paper uses mutual information to calculate the coherence level between consecutive characters and words, thus collect all possible candidate segments. This stage purely depends on the probalistic distribution of each pattern and its context, and requires no linguistic knowledge. Later I apply linguistic resources like character lists, Chinese classifier word list, and organization suffixes to assign the most probable category of each guessed segment; some simple syntactic knowledge like coordination and prepositional phrase-structure is also applied at this stage.

3 Entropy and Mutual Information

This section introduces the calculation the entropy of an arbitrary-length text segment and the mutual information between two patterns, which is the basis for next-step feature identification.

3.1 Statistics of the frequencies and contexts for any-length text segments

An input document is firstly segmented against a general-domain dictionary, whose words are already verified in a large corpus to be common-sense words. High-accuracy pattern-matching features like numbers, dates, URL and e-mail addresses are also identified to reduce overall token numbers and thus accelerate the next-step sorting procedure. But less-accurate features like some Chinese phrases "— X — Y" (where X and Y are single Chinese characters of a same part-of-speech. Such phrase pattern is designed to tackle phrases like "一草一木" and "一瘸一拐") are disabled, because recognizing such phrase patterns sometimes causes segmentation errors in its nearby context, and thus

causes more side effects; rather these patterns are recognized after the identification of proper names, unknown words, and document terms.

The sorting procedure uses the same algorithm as in Guo (1996). The sorting algorithm is quite suitable for sorting large quantity of patterns. For n text patterns, this algorithm finishes in $O(n*log(n))$ time while costing $O(n)$ memory.

Sorting results look like this:

Left context	pattern
.....	.....
我/国/小将/	常/昊/今天/表现/
/刘/小\光/和/	常/昊/均/在/中盘/
中国/的/.../、/	常/昊/六/段/和/
第/二/大/	城市/釜山
.....	.....

Table 1: appearance of "常/昊"

In the sorting result sequences, each pattern is associated with the number of identical tokens to its immediate succeeding pattern. For example, in Table 1, the associated number would be 2, 2, 2, and 0. With these associated numbers, it's very easy to calculate the occurrence of any pattern, ranging from a single token to any-length of tokens. To calculate the occurrence frequency of the single token "常" in table 1, just locate its first appearance in the sorted sequence, then count all succeeding patterns with an associated number of no less than its length 1, which is the number of tokens it has, we'll know its occurrence frequency is 3. For a 2-token pattern "常/昊", locate its first appearance and count all succeeding patterns with an associated number of no less than its length 2, we'll know its occurrence frequency is also 3.

A pattern's right context, in sorted order, can also be directly obtained from the sorted sequences. For example, the immediate right context of "常/昊" are "今天", "均/在", and "六/段", all with an occurrence frequency of 1. I augmented the sorting algorithm so that each pattern's left contexts can also be quickly obtained in sorted order.

3.2 Entropy and Mutual Information

Following the method of using mutual information to construct a relevance network in Butte (2000), this paper defines the mutual information (MI) between two adjacent text segment s and t as

$$MI(s,t) = H(s) + H(t) - H(st) \qquad (1)$$

where H(s) and H(t) are the entropy of s and t in the whole document; H(st) is the entropy of the text segment concatenated by s and t, which indicates how often s and t appear adjacently in the document.

The entropy H(x) of a text segment x is defined as

$$H(X) = -\sum_{i=1}^{n} p(x_i)\log_2(p(x_i)) \qquad (2)$$

Since the occurrence frequencies of characters and words in a document are disperse values, I use the frequencies that a pattern appears in the paragraphs of a document. In other words, a document is firstly segmented into paragraphs, and then a pattern's frequencies in each paragraph are counted. For example, in a document about a new medicine "糖脉康" for curing diabetes, there are altogether 8 paragraphs. Among these paragraphs, "脉" -- as a single segmentation unit -- appears totally 7 times in 5 paragraphs, namely 1, 1, 2, 2, 1; thus its probabilities are 0.14, 0.14, 0.28, 0.28 and 0.14. By formula (2), its entropy is 2.24.

The entropy numbers calculated for each segmentation unit (either a word or a lone character) via this method implies each segmentation unit's ability of forming a longer feature. A larger entropy number means that the segmentation unit is used in all paragraphs of the document more equally, and thus having a larger possibility of forming new features such as document terms, unknown words, or proper names. Another alternative to calculate each segmentation unit's entropy might be based on their occurrence frequencies in each sentence, but since a document contains much more sentences,

the occurrence frequencies are usually very small, leading to all probabilities are almost the same; the result of the overall method is worse than using occurrence frequencies in paragraphs.

Table 2 shows the entropy numbers of patterns "糖*" – patterns starting with "糖" -- in the above-mentioned sample documents:

Paragraph id	*Left context*	*The focused pattern* "糖*"
17	不仅/降低/血	糖/，
14	/化学/合成/降/	糖/药/对/糖尿病/并发症/
2	/已/有/的/降/	糖/药/，副作用/
3	/的/新/药/"/	糖/脉/康/"/。
22	预计/明年/"/	糖/脉/康/"/的/产值/
23	，/"/	糖/脉/康/"/的/推广/和/
19	/认为/"/	糖/脉/康/"/的/问世/
20	，/对/"/	糖/脉/康/"/的/有效/成分/
21	为/增强/"/	糖/脉/康/"/这/类/面向/二十一世纪/的/
16	"	糖/脉/康/"/颗粒/不仅/降低/
7	/百分之九十/以上/为/	糖尿病/ＩＩ/型/。
10	/到/二十一世纪/，/	糖尿病/ＩＩ/型/将/
8	<begin of paragraph>	糖尿病/ＩＩ/型/其/并发症/如/
4	/有助于/抑制/	糖尿病/ＩＩ/型/在/……/的/流行/。
15	合成/降糖/药/对/	糖尿病/并发症/均/
18	，/对/	糖尿病/并发症/有/特殊/疗效/，
2	/提取/出/对/	糖尿病/有/特殊/疗效/的
	……	

Table 2: patterns formed by "糖*" and its contexts

Based on the occurrence frequencies in Table 2, the entropy numbers of patterns like "糖*" are calculated. Table 3 lists the calculation result:

Pattern and its entropy value	remarks
糖2.30	"糖脉康"7次，"降糖药"2次，不计在"糖尿病"中的14次
脉2.24	全部出现在"糖脉康"中，7次
康2.24	全部出现在"糖脉康"中，7次
脉,康2.24	全部出现在"糖脉康"中，7次
糖,脉/康2.24	7次

Table 3: entropy numbers of patterns "糖*"

The mutual information between two text patterns *s* and *t* indicates how tightly they are used as a whole unit, which is taken as a criterion of the semantics coherence level of these two patterns. In this way, mutual information can be used as a metric between two patterns related to their degree of independence. It's hypothesized that the higher mutual information between two patterns, the more likely they form a coherent unit.

A mutual information at zero means that the two patterns *s* , *t* and their combination occurs only once in the document, thus gives no meaning in statistics, and any possible features formed by *s* and *t* can not be recognized by this paper's document-statistical method.

A higher mutual information means that one pattern is non-randomly associated with the other; in semantics, this means they form a new concept or phrase. In the sample document, since the two words "脉" and "康" are always used together, and leads to a very high mutual information value of 2.24, only slightly less than the mutual information value 2.30 of "糖" and "脉".

Since the calculation of mutual information has already taken into account the occurrence frequencies and distributions of the segmentation units, it's more suitable to act as the criterion of detecting words coherence level. From the sorting result, I used to select those text segments with

two or more occurrences as candidates of new features. But the occurrence number doesn't tell the coherence level of a segment's constituents, and neither is it a good metric to decide how long a text segment extends. Some of my experiments prove that using mutual information is better than using conditional probability or using frequency or distribution alone. Also due to the fact that mutual information values are compared within a document, there seems no need to normalize the values.

4 Algorithm of Feature Identification

After calculating the mutual information values of all consecutive segmentation-unit pairs, I use the following three steps to identify new features in the document:

(1) select those coherent token pairs t_i and t_{i+1} whose mutual information values are above a threshold; the selected pairs are used as "seed" for extension in the followed step;

(2) scan those tokens following the pair [t_i and t_{i+1}] to determine its right boundary of the new feature based on mutual information values of a segment and its extensions. Similarly scan the pair's leftward context to determine its left boundary;

(3) assign a category (proper name, unknown word, or document term) for each of the sifted candidate patterns, and also compute a confidence value of the pattern as the assigned category.

4.1 Construction of candidate feature pairs

To determine what adjacent segmentation-unit pairs can be part of a new feature, this paper uses a relative number of all mutual information values as the threshold. Using a larger threshold will reduce the set of the candidates, leading to a relatively high accuracy rate but a low recall rate; while using a smaller threshold leads to low accuracy rate and high recall rate. As a trade off, I assign the threshold as a relative percentage of the maximum of all mutual information values, i.e.

Threshold = k * MAX (all mutual information values)

In experiment, 0.6 is used as the coefficient k.

While selecting possible candidate features from bigram pairs, some morphological cues can be also used. For example, in the above-mentioned sample document, "对/糖尿病" has a high mutual information value, due to the fact that "对" always precedes "糖尿病" in this document. From a common dictionary and a simple part-of-speech tagging engine, it's easy to know that "对" is almost definitely a preposition when used before a noun and forms a prepositional phrase, and thus such patterns could be eliminated to reduce noises for later steps and improve accuracy rate. Patterns with punctuation marks are also eliminated from the candidate set.

4.2 Determination of the boundaries of a candidate feature

To identify those possible new features consisting more than two segmentation units, I use the candidate pairs obtained from previous step as "seeds", and then extend the seed to both right and left. By comparing the mutual information values before and after an extension, the algorithm determines one of these three actions: (1) to stop extension, i.e., only adopt the original short pattern as a candidate feature; (2) to adopt both patterns as candidate features; or (3) to adopt the extended pattern and eliminate the original shorter pattern.

Extensions are tried in both directions to determine both the right and the left boundaries of a possible candidate new feature.

For the selected pair pattern " t_i t_{i+1} ", following methods are applied to determine its right boundary:

- If all followed tokens has only one occurrence, the pattern will not be extended, and keep itself as a candidate of new feature. In the example "常/昊" in Table 1, there are three possible extended patterns, namely "常/昊/今天/", "常/昊/均/" and "常/昊/六/"; all of them appears only one time in the document, therefore only accept "常/昊" as a candidate feature, and neglect any of the extended three patterns. The rightward extension of this pattern stops as a result;

- If a pattern has only one followed tokens, then eliminate the original shorter pattern and extend it to the extended longer pattern. In the example "糖尿病/ I I /", all the followed tokens are "型", therefore take "糖尿病/ I I /型" as a candidate feature and at the same time eliminate "糖尿病/ I I ";
- If some of the extended patterns occur more than once, then compute mutual information values for all of them. Then compare the original mutual information $MI(t_i,t_{i+1})$ with that of the extended pattern $MI(t_it_{i+1},t_{i+2})$; divide the comparison result into three types:
  a) For cases of $MI(t_it_{i+1},t_{i+2}) < \lambda_1 MI(t_i,t_{i+1})$, λ_1= 0.4 in actual experiment, which means all the extended patterns have a lower coherence level, then stop rightward extension for "t_i/ t_{i+1}", and keep it self as a candidate new feature;
  b) For cases of $MI(t_it_{i+1},t_{i+2}) > \lambda_2 MI(t_i,t_{i+1})$, λ_2= 0.9 in actual experiment, which means the extended patterns have a strong coherence level, then eliminate the original pattern "t_i / t_{i+1}", adopt the new longer pattern "t_i / t_{i+1} / t_{i+2}" into candidate set, and continue to check its rightward patterns for possible extension;
  c) For cases of a intermediate mutual information of the extended pattern, i.e., the value of $MI(t_it_{i+1},t_{i+2})$ falls between the range of $\lambda_1 MI(t_i,t_{i+1})$ and $\lambda_2 MI(t_i,t_{i+1})$, then keep the original pattern "t_i/ t_{i+1}" in the candidate set, adopt the new longer pattern "t_i/ t_{i+1} / t_{i+2}", but stop checking any possible rightward extension for "t_i/ t_{i+1} / t_{i+2}".

Use the similar way to find the pattern's left boundary.

As an additional syntactic rule, punctuation marks and some kinds of empty words like particles and prepositionals also stop the extension. For example, in the above example in Table 2, "糖/脉/康/" is followed by a quotation mark, thus "糖/脉/康/" is taken as a new feature; Similarly, "糖尿病/ I I / 型" is followed by a full stop punctuation mark in the 7th paragraph, which gives a strong clue to stop extension for its rightward boundary, although in other sentences it's following by various kinds of words.

4.3 Assignment of a category and confidence to the candidate features

All candidate features identified from the previous steps are divided into three categories: proper names (person, place, organization, abbreviations, trade marks, etc.), unknown words, and document terms. The following rule-based heuristic clues are used to determine the categories that a feature belongs to:

- Whether the feature is made up of lone characters or it is made up of words:
  A feature that's made up of lone characters are often person name, place name, abbreviation, or an unknown word; and a feature of words are often organization names or document terms;
- Whether the feature has a proper name suffix:
  Proper name suffixes, such as place name suffix, organization suffix, and personal titles, are already collected in our linguistic resources. These suffixes can help to determine the category of a new feature. In formal texts, most proper names occur together with such a strong clue for at least one time in the whole document: a person name often occurs with his/her title like "President XYZ" and "Professor XYZ", an organization's full name is used for the first time before using its partial names like "四川富益电力股份有限公司" (*Si-Chuan Fu-Yi Electrics Company*) and "四川富益" (*Si-Chuan Fu-Yi*);
- Commonly used character tables of proper names:
  We've collected commonly used characters for Chinese person names, Chinese place names, and transliterated foreign person and place names. These tables are used to check features of lone characters. For example, if the first character in a three-character feature is a Chinese family name and the other two characters are found in Chinese person name character table, then this is taken as a strong evidence to indicate that the feature is a Chinese person name;

- Clue words around the feature:
  Often there're some clue words to help to determine a feature's category. For person names, clue words can be human occupation names like "记者" (reporter); in other words, if such a clue word is found before or after a feature, this feature will be have a higher confidence of being a person name. Chinese organization names are often preceded with a place name that they're owned like in the previous example "四川富益电力股份有限公司", thus place names can act as clue words for organization features.

For the feature of abbreviations, I now only focus on those of organization names. In Chinese, most of the shortened forms are either a part of a full name, like "四川富益" (*Si-Chuan Fu-Yi*) and its full form "四川富益电力股份有限公司" (*Si-Chuan Fu-Yi Electrics Company*), or an acronym made up of several characters selected from its composite words, like "川富电" (*Chuan Fu Dian*) for the same full name. The first kind of shortened form is somewhat like the procedure of a rightward pattern extension. For the second kind of shortened form, it's easy to retrieve the words that each character stands for, from the statistics obtained at the very initial step. Both these two kinds of shortened forms reference to their full form, and are stored as a canonical-variants group.

Even after applying such clues, if there're still some features that are not determined with any categories, those made up of lone characters are assigned as unknown words, and those made up of words are assigned as document terms.

If there're too many unknown categories, the document might be of another domain. The candidate features are clustered for manual review; from the review result, domain-specific rules summed up to act as domain-specific knowledge.

The steps of determining a feature's category also helps to determine its confidence. Strong clues that determine a feature's category are often used as strong confidence of the feature itself. Together with factors of the occurrence frequency and the trail of a feature's mutual information extension, each feature is assigned a confidence value.

5 Experimental Results

The algorithm can be evaluated in two aspects: its ability to collect candidate features, and the quality of categorizing these new features. To improve the accuracy and recall rate of collecting candidate feature is the main aim of introducing mutual information into the whole algorithm.

I use UPenn Chinese treebank as a test corpus. The corpus contains about 180 thousands Chinese characters, or 99.7 thousands words. Among all words there are 9700 proper names, mainly names of persons, places and organizations. Following is a sample of UPenn treebank markup:

```
(NP-OBJ (NP-APP (CP (WHNP-1 (-NONE- *OP*))
                    (CP (IP (NP-SBJ (-NONE- *T*-1))
                            (VP (PP-DIR (P 对)
                                        (NP (NN 糖尿病)))
                                (VP (VE 有)
                                    (NP-OBJ (ADJP (JJ 特殊))
                                            (NP (NN 疗效))))))
                        (DEC 的)))
                (ADJP (JJ 新))
                (NP (NN 药)))
        (NP-PN (PU “)
               (NN 糖脉康)
               (PU ”)))
```

Of all the features targeted by this paper, four types – namely, person names, place names, abbreviations and unknown words -- roughly map to the "words" in UPenn treebank, while the other two types -- document terms and organization names -- roughly map to the "phrases" in UPenn treebank. For example, "糖/脉/康/" is an unknown word, "糖尿病/并发症" is a document term in this algorithm's result; both of them match ideally with the word "糖脉康" and the noun phrase "糖尿病并发症" in the treebank.

While measuring the algorithm's precision rate, an identified feature is considered correct if it matches UPenn treebank's any bracket level. In this way the precision rate is 87%.

To measure the algorithm's recall rate, we must first decide what are "correct features", whereas the bracket level in UPenn treebank varies quite a lot it their length, and doesn't quite fit. If we limit the set of "correct feature" to be the treebank's innermost bracket level, the recall rate is above 90%. But such measure will only cover a small portion of the identified document terms; for example, "新药" is identified as a document term, but UPenn treebank tags "有特殊疗效的新药" as a whole phrase, instead of desired level "新药". To expand the "correct features" to noun-phrase level of UPenn treebank will reduce the algorithm's recall rate to only 42%.

Among the precisely identified features, the overall category tagging precision is above 60%.

To test the algorithm's adaptability to other domains, I selected several English documents. The recall of organization names is 86%, indicating that the algorithm's ability to collect candidate features keeps the same level as in the UPenn Chinese treebank experiment.

6 Conclusion and Future Work

This paper uses entropy and mutual information to collect candidate features, and then combines with existing linguistic resources such as proper name suffixes and character tables. It proves to be more robust and efficient than the original methods of sentence-based feature identification. Also, because it works on the whole document, a clue at only one place will also take effects to other places; this is especially important in identifying abbreviations and coreferences.

Since this method tries to use document statistics to find candidate features, it's not applicable for those features that only appear one time in the document, for example, the person name "/刘/小/光/" in Table 1. To produce an overall segmentation result for the whole document, conventional methods like joining lone characters based on person name character table must also be applied. Simple syntactic rules can also be applied, e.g., in a simple coordination structure like "A、B、和C", if one piece B is already confidently identified as a person name, then other pieces A and C are most possibly person names. To find a good strategy to make different engine-modules interoperate well is also a challenge.

References

Baluja S, Mittal V, Sukthankar R. 1999. Applying machine learning for high performance named-entity extraction. Proc. Pacific Association for Computational Linguistics. 1999.

Butte AJ, Kohane IS. 2000. Mutual Information Relevance Networks: Functional Genomic Clustering Using Pairewise Entropy Measurements. Pacific Symposium on Biocomputing (PSB 2000).

Chen Xiaohe. 1999. A Total Solution for Unknown Words Identification in Automatic Segmentation. *Journal of Language Applications*, 1999(3).

Guo Zhili, Yuan Chunfa, Huang Changning. 1996. An Approach to Determine the Boundary of Chinese *de*-phrase. Proceedings of the International Conference on Chinese Computing (ICCC-1996).

Ji Heng, Luo Zhensheng. 2001. Inverse Name Frequency Model and Rules Based Chinese Name Identifying. Proceedings of the sixth Joint Symposium on Computational Linguistics (JSCL-2001).

Sun Maosong, Shen Dayang. 1995. Automatic Identification of Chinese Person Names. *Journal of Chinese Information Processing Society,* 1995(2).

Tan Hongye. 1999. Study on the Automatic Identification of Chinese Place Names. Proceedings of the 5th Joint Symposium on Computational Linguistics (JSCL-1999).

Zhang Xiaoheng. 1997. Identification of Chinese Organization names. *Journal of Chinese Information Processing Society,* 1997(4).