Design and Implementation of A Classifier for Chinese E-mails

Jim Z. C. Lai, and Wen-Feng Wu
Department of Information Engineering
Feng Chia University
100 Wenhwa Rd., Taichung, 40724 Taiwan
E-mail: lat@fcu.edu.tw, wwf@mail.vclab.iecs.fcu.edu.tw

ABSTRACT

Many E-mail clients allow their users to define rules for filtering. However, any user that keeps a large amount of E-mails will find it is difficult to maintain such a set of rules. This paper presents a mail classification system for Chinese mails. We use different classifiers for different feature sets, and combine some classifiers to classify the mails. We embed our email classifier in a mail server to tag mails with class labels. The users can category their mails directly according to class labels. This will reduce the burden of users to classify the mails and facilitate the use of setting those complex rules. Experiments show that the purposed system has its acceptable accuracy.

keyword: Email, information retrieval, document classification, Chinese processing

1. INTRODUCTION

E-mail has become an indispensable part in many people's life because of the popularity of the Internet. The amounts and types of mails increase day after day. In order to facilitate the searching when the user is looking for a previously received mail, many users organize their mails to different folders. Traditionally, the users filter their mails manually. Some E-mail applications allow the users to classify their mails by setting filtering rules. But those E-mail filters are based on keywords, and the user has to construct keywords manually. However, it may need experiences to set those complex rules. It's also not easy to analyze and to produce a set of rules that can represent the whole class precisely. Thus, methods for automatic classification are become necessary.

In order to facilitate the process of filtering, many researches apply techniques of machine learning to classify documents automatically in recent years [1,2]. In machine-learning approach, an E-mail classifier was trained automatically to classify mails. It greatly decreases the burden of the users to classify mails. In the previous studies on E-mail filtering, a mail is considered as a semi-structure data. Beside the textual information in the mail content, they also refer the text in specific headers for classification, for instance, “From”, “To” and “subject”. But most of those studies use textual information only [3,4]. There are other many characteristics of E-mails can be used to provide information for E-mail filtering.

Chinese documents are different from the corresponding English documents. In English, words are separated by space or symbols. In Chinese, a word might be composed of one, two, or more adjoining Chinese characters. Moreover, most words do not have explicit word boundaries. We cannot extract Chinese words simply like the way of extracting English words from a document. If we want the filter to have the ability of processing Chinese mails, we need a segmentation method to extract Chinese words.

The purpose of this paper is to design a filtering system for Chinese E-mail classification. By studying past researches on document classification and the methods of Chinese processing techniques, we try to construct an effective filtering system, which applies machine-learning techniques to classify mails.

2. SYSTEM ARCHITECTURE

In this paper, we incorporate the E-mail classifier into an E-mail gateway. The role of our experimental system, E-mail gateway, is shown in Figure 1. For the E-mail server, E-mail gateway is a filtering program that receives each mail before the mail server. Then, the E-mail gateway scans the content of each mail, and takes action according to the information found. Currently, there is only one kind of action in our system: tagging the mail a predicted categorization label and sending the mail to the E-mail server.

Figure 1. E-mail Gateway

The system consists of five distinct functional components. The first component is the mail receiving and delivering module. It provides the ability of connecting to other mail servers via ESMTF protocol.
The E-mail gateway server uses this component to receive mails from remote servers and send mails to the users on destination E-mail server. The second component is the feature extraction module, which extracts related features from incoming mails and feeds the next component, the classification module, with these features. The third component is the classification module, which consists of two classifiers: decision tree classifier and naive Bayes classifier. The classification module is used to predict the categorization of a mail. The fourth component is the mail categorization module. This component tags mails with categorization labels through modifying mails’ subjects. The fifth component, Chinese lexicon extraction module, is used to generate necessary Chinese lexicon.

In the following we will introduce feature extraction module and Chinese words extraction module in details.

### 2.1 E-mail Feature Extraction

There are two types of features used in our system: textual feature and structure feature. Textual feature consists of words in the text file. Structure feature is shown in Table 1. There are ten mail structure features: the first feature is “subject contain “re” ” indicating whether “re” occurs in the subject. The value of the first feature is either 0 or 1, where 1 stands for present and 0 stands for absent. The second feature is “subject contain “fw” ” indicating whether “fw” occurs in the subject. The value of the second feature is the same as that of the first feature. The third feature is the size of the whole mail and the mail size is counting in bytes. The fourth feature is the number of words in a mail and it is sum of Chinese characters and English words. The value of the fifth feature is the number of words divided by the number of symbols. The sixth feature is the number of mail receptors and we search receptors in two specific mail headers, “To” and “Cc”. The seventh feature is the number of images in the mail. The value of the eighth feature is the total size of images divided by the number of images. The ninth feature is the sum of the sizes of video files in the mail. The tenth feature is the sum of the sizes of audio files in the mail

The procedure of E-mail feature extraction is shown in Figure 3. In the beginning, the MIME (Multipurpose Internet Mail Extensions) parser will extract attached files from mails. If there are textual documents in E-mails, the document converter will be invoked to convert documents into text files. Currently, Html Converter and MS Word file converter are implemented in document converter. Structure feature extractor is used to generate E-mail’s ten structure features. Textual feature extractor is used to generate E-mail’s textual features. Our system can process data in two languages, English and Chinese. In English, it converts English words to lower case, and removes stop-words appearing in stop-list, and keeps word stems using porter’s algorithm. In Chinese, we use the word segmentation rules to extract words from Chinese sentences in documents.

### 2.2 Chinese Words Extraction

For Chinese language, the minimum meaningful unit is Chinese word, not Chinese character. If we use Chinese character as the index unit of information, the precision of classification will decrease dramatically. To perform Chinese words extraction, we use the word segmentation rule to extract word, and we also use a statistics-based method to construct a necessary lexicon database automatically. The reason why we use this approach is described in Section 4.
The procedure of Chinese lexicon construction is illustrated in Figure 3. The corpus used in the system is acquired directly from E-mails. At the beginning, Chinese word extractor counts the n-grams in the document and then keeps Chinese words based on some criteria. Not every high frequency n-gram is semantically accepted word. Hence, we add an optional user interface to help remove wrong words. Besides, we utilize an error word database to record those wrong words, which were deleted. Therefore we can avoid the possibility of extracting the same wrong words. After users' filtering, those wrong mistaken words will save in the database of mistake words automatically. At the end, users can import the newly extracted words to the Chinese lexicon database.

3. DOCUMENT CLASSIFICATION

In this section, we introduce the approach of document classification used in our system. We do not use all of features to train a classifier. We train difference classifiers with different types of features. A decision tree classifier, C4.5, is trained using structure features. Because the number of textual feature is quite large, and decision tree classifier is not easy to handle a large number of features, we use naive Bayes classifier for the task of text classification.

3.1 C4.5 Classifier

C4.5 is a widely used decision tree classifier in many applications of machines learning. It is an extension of another decision tree algorithm - ID3. C4.5 use a Gain Ratio instead of the Information Gain as the criterion of attribute selection. Gain Ratio avoid the bias that favors attributes with many values over those with few values. The Gain Ratio of an attribute is determined as follows:

\[
\text{Gain}(T) = \text{info}(T) - \sum_{i=1}^{s} \frac{|T_i|}{|T|} \times \text{info}(T_i).
\]

\[
\text{Split}(T) = - \sum_{i=1}^{s} \frac{|T_i|}{|T|} \times \log_2 \left( \frac{|T_i|}{|T|} \right)
\]

\[
\text{info}(T) = - \sum_{j=1}^{N\text{Class}} \frac{\text{freq}(C_j, T)}{|T|} \times \log_2 \left( \frac{\text{freq}(C_j, T)}{|T|} \right)
\]

where \(T_1\) through \(T_s\) are the s subsets of examples resulting from partitioning \(T\) by the s-valued attribute; \(N\text{Class}\) denotes the number of classes; \(\text{freq}(C_j, T)\) represents the number of examples in set \(T\) belonging to class \(j\); and \(|T|\) denotes the number of examples in set \(T\).

In order to avoid overfitting, we also use the error-based pruning approach to perform the function of post-pruning. The more detailed algorithm about C4.5 can be found in the reference [5].

3.2 Naive Bayes Classifier

Using the naïve Bayes classifier, we view documents as a bag of words and do not consider the words' positions in a document. A document is represented as a vector \(<N_{i1}, N_{i2}, \ldots, N_{im}>\), where each component \(N_{ij}\) is the number of times the word \(t\) occurred in the document. Let \(\{c_1, c_2, \ldots, c_k\}\) be a set of class labels of a class variable \(C_{MAP}\). \(\{d_1, d_2, \ldots, d_m\}\) is a set of training documents. The class label of a document \(d'\) is determined as follows:

\[
C_{MAP} = \arg \max_{C_j} P(C_j \mid d'), \ j = 1 \sim k,
\]

where

\[
P(C_j \mid d') = \frac{P(C_j)P(d' \mid C_j)}{P(d')}
\]

\[
P(d') = \sum_{j=1}^{k} P(C_j)P(d' \mid C_j)
\]

Naive Bayes classifier uses a naïve assumption: each word inside a document occurs independently, and in this paper, we adopt the multinomial event model [10]. Thus, the formulation of \(P(d' \mid C_j)\) is

\[
P(d' \mid C_j) = \prod_{t=1}^{V} \frac{\prod_{i=1}^{N_i} P(w_t \mid C_j)^{N_i}}{N_i!}.
\]

where \(|V|\) is the number of the unique word in training documents; \(N_i\) is the number of times word \(t\) occurred in document \(d'\); and \(P(w_t \mid C_j)\) is the probability of word \(W_t\) occurs in documents that belong to class \(C_j\). When value of \(P(w_t \mid C_j)\) is zero, \(P(d' \mid C_j)\) will have a
wrong value; and if some classes have very few training data, \( P_{(w_i|C_j)} \) will had an unfair value. Hence, we use a smoothing method, m-estimate [6], to avoid these biases. As mentioned above, \( P_{(w_i|C_j)} \) is determined as follows:

\[
P(w_i | C_j) = \frac{1 + \sum_{d=1}^{D} N_i P(c_i | d) | d_i |}{|V| + \sum_{d=1}^{D} |d_i|} \quad (8)
\]

where \(|V|\) is the number of the unique word in training documents; \(D\) is the number of training documents; and \(N_i\) is the number of times word \(t\) occurs in document \(d_i\).

One of the major characteristics of text classification is the high dimension of feature space. It increases the computational time of classification. A straightforward way is to reduce the number of features. Information Gain has performed well in previous studies [7,8], thus we use it as the criterion of feature selection. Let \(\{c_1, c_2, \ldots, c_m\}\) denote the set of classes. The Information Gain of the word \(t\), \(G(t)\), is determined as in Equation 9:

\[
G(t) = \sum_{i=1}^{m} \left[ P(c_i | t) \log P(c_i | t) \right] + \sum_{i=1}^{m} \left[ P(c_i | t) \log P(c_i | t) \right] + \sum_{i=1}^{m} \left[ \frac{P(c_i | t)}{P(c_i | t)} \right]
\]

In this paper, there are total four criteria of feature selection:

1. Pruning the word if its number of occurring in the training corpus is less than a threshold.
2. Pruning the word if the number of document that contains this word is less than a threshold.
3. Using the document as a counting unit and pruning the word that has low Information Gain.
4. Using the word as a counting unit and pruning the word that has low Information Gain.

### 3.3 Combination of Classifiers

The procedure of determining the class of a mail is described in the following. First, using the decision tree classifier to predict the class of a mail. The predicted class \(C_{\text{c45-max}}\) for the mail \(M\) is determined as follows:

\[
C_{\text{c45-max}} = \arg \max_{C_j} P_{C_{\text{c45}}}(C_j | M) \quad (10)
\]

where \(P_{C_{\text{c45}}}(C_j | M)\) denotes the probability of the class \(j\)-th class given E-mail \(M\); and \(n\) denotes the number of class. Ten structure features cannot provide sufficient information for classifying mails, unless the probability of the class \(C_{\text{c45-max}}\) is greater than or equal to \(TH_{\text{c45}}\) or the E-mail does not contain textual information, the E-mail would be classified as class \(C_{\text{c45-max}}\). If the probability of the class \(C_{\text{c45-max}}\) is less than \(TH_{\text{c45}}\), we use naive Bayes classifier to predict the class for each textual document in the E-mail. The class of a textual document \(d_i\) is determined as follow:

\[
C_{\text{NB}} = \arg \max_{C_j} (P_{\text{NB}}(C_j | d_i) | N_j |) \quad j = 1 \sim n \quad (11)
\]

where \(P_{\text{NB}}(C_j | d_i)\) denotes the probability of the \(j\)-th class given the textual document \(d_i\); \(n\) is the number of classes, and \(N_j\) is the information about whether the \(j\)-th class is a possible class for the E-mail. \(N_j\) is determined by the following formula:

\[
N_j = \begin{cases} 
0, & \text{if } P_{\text{c45}}(C_j | M) < TH_{\text{c45}} \\
1, & \text{if } P_{\text{c45}}(C_j | M) \geq TH_{\text{c45}} 
\end{cases}
\]

If there are \(m\) textual documents in a mail, there are \(m\) predicted classes at most. We simply regard the class that had maximum priority as the class label of the whole E-mail. Currently, the class priority is the same as the order of class number. As mentioned above, the class of a mail is determined ny one of the following three phases.

1. If the class that predicted by the C45 classifier has high confidence, the class of the E-mail will be determined in this phase. Otherwise go to the next phase.
2. Predicting class using the naive Bayes classifier and class information from Phase 1.
3. If the class of the E-mail cannot be determined in the above two phases, we assign the class predicted in Phase 1 to the E-mail.

### 4. CHINESE PROCESSING

Methods of Chinese phrase extraction can be roughly divided into two types: dictionary based approach and statistical based approach. Dictionary-based approach is also called rule-based approach. This kind of method consists of a lexicon and a set of word segmentation rules [7,11]. The lexicon is made in advance and inspected artificially, like the CKIP Chinese lexicon [9]. Word segmentation rules are utilized to solve the problems of segmentation ambiguity in Chinese word extraction. Statistical-based approach directly calculates the frequency of n-grams from a large amount of documents, and uses the mathematical method to determine if the n-grams are meaningful Chinese words for human. In this paper, N-gram means a string connected by \(n\) continuous Chinese characters only.

Based on the above reasons and our requirements, we decide to use a hybrid approach. That is, use statistical-based method to extract phrases to construct a lexicon for the system, then, using constructed lexicon and word segmentation rules to perform the function of Chinese processing. We use the net frequency [10] as our
evaluation function to determine if a string is meaningful. Reference [10] used a document once at a time to extract Chinese words. It had unstable performance. For this reason, we use a set of documents that belong to the same class once at a time to extract Chinese words. However, it produces a problem about efficiency. Searching the string that has some prefix and suffix is an expensive part in the original algorithm. Hence, we propose an implementation based on the efficiency-improved version. The improved algorithm we proposed for determining net frequencies of words will be described in the following paragraph.

First, we introduce the symbols and definitions used in the algorithm.

1. Let $C$ denote a string that consists of $n$ adjacent Chinese characters, that is, $C= c_1, c_2, ..., c_n$, where $c_i$ denotes a Chinese character. Let $C_L$ denote the longest left substring of $C$ with length $n-1$, where $C_L = c_1, c_2, ..., c_{n-1}$. Denote $C_R$ as the longest right substring of $C$ with length $n-1$ and $C_R = c_n, c_{n-1}, ..., c_2$. Let $C_{Mid}$ be a sub-string of $C$ with length $n-2$, where $C_{Mid}$ lacks of the leftmost and the rightmost characters of $C$ and $C_{Mid} = c_2, c_3, ..., c_{n-1}$. $|C|$ denotes the length of string $C$. In this case, $|C| = n$.

2. $Ngram-Table$ denotes a list that consists of all possible patterns occurred in the corpus and corresponding frequencies. The form of each item in $Ngram-Table$ is $(C, F_C)$, where $C$ denotes a pattern and $F_C$ is the corresponding frequency of pattern $C$ occurred in the corpus. We use $NgramF(C)$ to denote pattern $C$'s corresponding $F_C$ in the $Ngram-Table$.

3. $Net-Table$ denotes a listing that the format of each item is same as $Ngram-Table$. We use $NetF(C)$ to denote the corresponding $F_C$ of pattern $C$ in $Net-Table$. When the algorithm is finished, $Net-Table$ would record all possible Chinese words and corresponding net frequencies.

4. Calculate $NetF(C)$. If pattern $C$ does not exist in $Net-Table$, we copy the item $(C, F_C)$ from $Ngram-Table$ to $Net-Table$ first.

5. $F_{TH}$ denotes a threshold of net frequency and we simply regard patterns as phrases, if their net frequencies are greater than $F_{TH}$.

The main algorithm is described as follow:

1. Extract all the patterns from the corpus and accumulate their frequencies. Each pattern and its frequency are stored in the $Ngram-Table$.

2. Do the following operations for each pattern $C$ in $Ngram-Table$.

   2.1 If $NgramF(C) \geq F_{TH}$ and $|C| = 3$, then, do
      
      $\text{NetF}(C_L) = \text{NetF}(C_L) - NgramF(C)$,
      $\text{NetF}(C_R) = \text{NetF}(C_R) - NgramF(C)$.

   2.2 If $NgramF(C) > F_{TH}$ and $|C| > 4$, then, do
      
      $\text{NetF}(C_{Mid}) = \text{NetF}(C_{Mid}) + NgramF(C)$.

   2.3 If $NgramF(C) < F_{TH}$ and $|C| > 4$, then, do

Because the corpus at each time is different, we determine the net frequency threshold dynamically. In this paper, the net frequency threshold $F_{TH}$ is set to $\log (N) + 2$, where $N$ denotes the number of Chinese characters in the corpus. In addition, in order to provide fast data retrieval, we use the hash-based method to manipulate $Ngram-Table$ and $Net-Table$.

After constructing the lexicon, we need segmentation rules to extract Chinese words. According to the previous study [11], when the lexicon is complete, the most frequently used segmentation rule, maximum match, has 95% accuracy, even it is used individually to perform word segmentation. So we simply use the maximum match rule to extract words. If there are several equal-long words existing in a sentence, the priority is the former one.

5. EXPERIMENTS

In the experiment, there are total 12 classes and 3319 E-mails. Except for the first class, all E-mails were collected from the personal mails. The data source of the first class is a set of sexual articles collecting from the Internet. We convert the format of these articles to the standard E-mail message. All experimental data is divided into two portions: training data and testing data. The distribution of the training data and testing data in each class is shown in Table 2.

<table>
<thead>
<tr>
<th>Class No.</th>
<th>Class Name</th>
<th>Training Data</th>
<th>Testing Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Pornography</td>
<td>149</td>
<td>22</td>
</tr>
<tr>
<td>2</td>
<td>Sex Advertisement</td>
<td>77</td>
<td>17</td>
</tr>
<tr>
<td>3</td>
<td>Software Advertisement</td>
<td>93</td>
<td>18</td>
</tr>
<tr>
<td>4</td>
<td>Politics &amp; Economics</td>
<td>377</td>
<td>51</td>
</tr>
<tr>
<td>5</td>
<td>Investment &amp; Finance</td>
<td>111</td>
<td>28</td>
</tr>
<tr>
<td>6</td>
<td>Health care</td>
<td>471</td>
<td>24</td>
</tr>
<tr>
<td>7</td>
<td>Computer Information</td>
<td>397</td>
<td>54</td>
</tr>
<tr>
<td>8</td>
<td>Joke</td>
<td>212</td>
<td>23</td>
</tr>
<tr>
<td>9</td>
<td>Entertainment</td>
<td>97</td>
<td>12</td>
</tr>
<tr>
<td>10</td>
<td>Personal Communication</td>
<td>417</td>
<td>52</td>
</tr>
<tr>
<td>11</td>
<td>Other Commercial Advertisement</td>
<td>503</td>
<td>66</td>
</tr>
<tr>
<td>12</td>
<td>Others</td>
<td>32</td>
<td>6</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td>2936</td>
<td>383</td>
</tr>
</tbody>
</table>

We use 2936 training data to train classifiers and 383 testing data to test the accuracy rate of trained classifiers. The accuracy rate of classification is 86%. The recall and precision rate for each class is shown in Table 3, where
recall and precision are defined as follows:

recall = \frac{\text{number of correct positive predictions}}{\text{number of positive predictions}}

precision = \frac{\text{number of correct positive predictions}}{\text{number of positive examples}}

Table 3 Recall and Precision

<table>
<thead>
<tr>
<th>Class No.</th>
<th>Recall</th>
<th>Precision</th>
<th>Class No.</th>
<th>Recall</th>
<th>Precision</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>100%</td>
<td>84.6%</td>
<td>7</td>
<td>88.8%</td>
<td>81.3%</td>
</tr>
<tr>
<td>2</td>
<td>58.8%</td>
<td>83.3%</td>
<td>8</td>
<td>100%</td>
<td>92%</td>
</tr>
<tr>
<td>3</td>
<td>61.1%</td>
<td>64.7%</td>
<td>9</td>
<td>91.6%</td>
<td>100%</td>
</tr>
<tr>
<td>4</td>
<td>100%</td>
<td>96.2%</td>
<td>10</td>
<td>82.6%</td>
<td>89.5%</td>
</tr>
<tr>
<td>5</td>
<td>96.4%</td>
<td>100%</td>
<td>11</td>
<td>75.7%</td>
<td>76.9%</td>
</tr>
<tr>
<td>6</td>
<td>91.1%</td>
<td>86.1%</td>
<td>12</td>
<td>33.3%</td>
<td>50%</td>
</tr>
</tbody>
</table>

Avg. Accuracy Rate: 86%

Table 4 Comparisons on Difference Classifiers

<table>
<thead>
<tr>
<th></th>
<th>NB</th>
<th>C4.5</th>
<th>C45+NB (Combination)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Test Accuracy Rate</td>
<td>79%(75%)</td>
<td>75%</td>
<td>86%</td>
</tr>
</tbody>
</table>

Table 5 Ratio of Class Determination phases

<table>
<thead>
<tr>
<th>Testing Data</th>
<th>Phase 1: C4.5 (High Conf.)</th>
<th>Phase 2: NB (Max. Prob.)</th>
<th>Phase 3: C45 (Max. Prob.)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>11.2%</td>
<td>86.6%</td>
<td>0.2%</td>
</tr>
</tbody>
</table>

Table 6 Experiment Statistics

- Number of Chinese Characters in Training Data: 4,658,084
- Number of Distinct Extracted Chinese Words: 19,315
- Number of Distinct Extracted Words: 34,268
- Number of Distinct Selected Features (Words): 7,232
- Number of Decision Tree Rules: 120
- Number of Decision Tree Nodes: 239
- Ratio of Testing Mails Contain Text Information: 96.4%
- Total Size of Training Data: 456,423,417 byte

6. CONCLUSIONS

In the paper, we present an experimental system for E-mail classification. We utilize both the characteristics of E-mail structure and textual information to model the E-mail classifier. We find that using the E-mail structure features can be helpful for classification. It also enables the ability of handling mails containing no textual information. In order to adapt to different attributes, we use different classifiers. Moreover, we combine two classifiers to make a more efficient classifier. The proposed system can process the E-mail that contains English words and Chinese words. It works even without offering Chinese lexicon in advance. Furthermore, we refine the Net Frequency algorithm and make it more efficient when handling a large amount of documents.

REFERENCES