行政院國家科學委員會專題研究計畫　成果報告

海關資料分類系統

計畫類別：個別型計畫
計畫編號：NSC94-2213-E-032-009-
執行期間：94 年 08 月 01 日至 95 年 07 月 31 日
執行單位：淡江大學資訊工程學系

計畫主持人：蔣定安

報告類型：精簡報告

處理方式：本計畫可公開查詢

中　華　民　國 95 年 10 月 17 日

Test Classification is functioned to categorized the unclassified text documents into the pre-defined category. We can take the customs export cargo for example and we chiefly make the comparison for the subjective cargo description of the texts and terms categorized by the typists. Most of the said fields are composed of free content, namely, the contents categorized by identical typists. During various periods, it will also exist in distinct cargo descriptions for the same cargo commodity. This situation cause poor classification results. It is necessary for users to lots of time and effort for analysis, filter and determination with the categorized data finished. Thus, we propose an improved term algorithm allowable for the notion of multi-classification and class priority. It is helpful to enhance the accuracy of existent classification system with the ultimate goal that it is available for users to perform rapid an accurate information inquiry.

Keywords: Text Categorization, Text Mining, Multi classification.

INTRODUCTION

Classification is a useful learning kit frequently used by mankind. It is mainly based on information context by following specific principles and places the unclassified information into the pre-defined categories. During the earlier period, it adopted with manual classification only by humane experience. Yet, nowadays, it is accessibly adopted with computers for automatic classification and processing operation. There are numerous methods proposed [1] and referred to as the technology of text classification.

This article takes "key words" for clues and uses the well-established database to compare the input data; and then, the texts with the same key words are selected for an identical category.

The experimental subjects are resourced from the B/L of customs cargo. The data format is viewed as a record for each batch of cargo within the database. The contents within documentary fields are all composed of English, numbers and symbols classified into company names, commodity numbers, commodity description and delivery dates, etc. Among them, the most important filed falls on the description for cargo. This field is filled with the descriptive texts classified by from the subjective discretion from typists to describe the cargo contents, such as "538 FROZEN MEAT; NET WEIGHT: 17462.4KGS, NET WT: 38,520.0000 LB". Thus, most texts of this field belong to format of free content, namely the contents classified by identical typists. During various periods, it will also exist in distinct cargo descriptions for the same cargo commodity.

We take the ROC China I/O Cargo Classification List (briefed as CCC code) for the classification reference, totally divided into 21 major categories and 97 sub-categories. Within the general introduction of CCC Code, it is mentioned that the cargo classification is framed with reference to The Harmonized Commodity Description and Coding System (briefed as HS) regulated by the World Customs Organization (WCO), issued in 1996. It is divided into 21 categories, 97 chapters (chapter 77 emptily reserved available for international revision in the future), 1214 sections (4-digit code) and 5113 items (6-digit code).

For example, the 1st major category is "Live Animals and Animal Products" belonging to the 1st living sub-category of living animal cargo. According to the experimental rules and the classification description of CCC coding, we extract the key words shown as Fig. 1：

CCC Code	Commodity	Keyword
0101.11.00.00-5	Live horses, pure-bred breeding animals	Live horse
0101.19.00.00-7	Live horses, other than pure-bred breeding animals	
0101.20.00.00-2	Live asses	Live ass
0101.20.00.00-0	Live mules and hinnies	Live mule
0101.20.00.00-0	Live mules and hinnies	Live hinny
0102.10.00.00-5	Live bovine animals, pure-bred breeding animals	Live bovine animals
0103.10.00.00-4	Live swine, pure-bred breeding animals	Live swine

Fig. 1

Within the "CCC Code", the first 4 digits are sectional codes initiated with "01" and belonging to the 1st sub-categories. The description shown in graph 1.1 all belong to the 1st subcategory. The "commodity" filed is the commodity description and exemplifications regulated by CCC code entirely composed of English strings. The commodity field of customs B/L is also composed of English strings. Base on such a feature, this article will focus on the "commodity" field listed by CCC code to extract the key words. Furthermore, we can make the analogy comparison between the said key words and training data. When the system is executed with the analogy comparison between key words and the commodity field texts within customs B/L, whatever the fields are classified with "Live horse" or "Live mule" will be classified into the 1st subcategory. If the key words are repeatedly extracted, such as the "Live horse" listed in 0101.11.00.00-5 和 0101.19.00.00-7, it will be omitted. Whatever it is the "Live horse" of 0101.11.00.00-5 and

0101.19.00.00-7 both belong to the 1st subcategory.

However, the simple comparison will meet with the problem whether the string lengths are the same or not:

1. The different lengths of key words:
For example, there is 2 word- sheep within the 1st sub-category (Live animals) of category1; and the 2nd sub-category (meat and edible meat offal) of category 1 is listed with a key word- sheep meat. If there is a record with the description of "538 FROZEN SHEEP MEAT; NET WEIGHT: 17462.4KGS, NET WT: 38,520.0000 LB". When we are making the comparison, the description listed with a word- sheep shall be classified into the 1st sub-category, but the description listed with "sheep meats" shall be wrongly put into the 2nd sub-category to cause the classification error.

2. The identical lengths of key words:
For example, if the key word-jack is frequently listed within the field of commodity name, it usually represents the "connector of household appliance" belonging to the sub-category 84; and there is another key word-jacket belonging to sub-category 62. If we adopt the "Fully Match" methodology to make comparison, whenever the key word is written as "jackets" in its plural form, thus this record will be omitted to lose its comparability and error tolerance. Accordingly, the data processing is available for porter stemming algorithm [2, 6] and each English word shall be returned its original singular form. However, there are still some occasional errors unavoidable; for example, the typists could possibly input the wrong words.

Thus, although the term database comparison is conveniently operable, yet it is mainly defected with low accuracy of classification. Seeing from the research results made by M.E. Maron, among 145 articles with 60 articles of no key words previously excluded, the comparison accuracy merely reaches 51 %.

Thus, this article is aimed to establish a classification system from the specific input data ("Cargo" B/L with overall English text input). It is based on the comparison methodology to propose 2 notions of "priority weight setting for categories" and "multi-classification" to improve the classification methodologies.

Regarding "priority weight setting for categories", if it exists in various wordy lengths, we can allow the longer words with higher priority weighting to avoid the possibly happening errors. For example, between "sheep" and "sheep meat", the "sheep meat" is obviously right. If the key words are identical, from the experiments, we can find that some categories will naturally pose higher priorities than other categories. We can tale "wood" and "chair" for example, the "chair" belonging to "furniture category" originally with higher priority. If we can combine the rules for priority setting into the classification system, it will naturally increase considerable accuracy. However, with the dynamic combination of rules for priority setting, it will cause some problems as what we call cycle priority. In chapter 3, we will discuss this problem in detail and also propose the "the algorithm to detect redundancy cycle priority" to sole the said problem.

The notion of multi-classification originates from the fact that some B/L is composed of more than 2 kinds of commodity, such as "428 FROZEN FISH AND FROZEN BEEF". Thus, if this kind of B/L is classified into a category, it seems highly unreasonable. Thus, the mechanism of multi-classification is allowable for systemic multi-classification in data processing.

The research resulting within this article suggests that after the priorities added and multi-classification allowed, within the B/L classification exclusive of the B/L of no Match key word, the classification accuracy can reach 95% above.

RELATIVE WORKS

M.E. Maron [5], in 1961, published his dissertation to be the pioneer to investigate the science of text classification. He proposed that by using clue words, we can make some available clues for text classification with much contribution for the subsequent scientific research about text classification. The sources of his research lectures for text classification were adopted from the periodical – Transaction on Electronic Computers with the document profile excerpt from the dissertation abstract within the said periodical. M.E. Maron totally selected 405 scientific dissertations with 260 articles to serve as training data and the remaining 260 articles were used for testing. All the articles were classified into 32 categories.

The dissertation of Hamill and Zamora [3] was proposed in 1980. The article texts were sourced from Chemical Abstracts with the document profile excerpted from the dissertation titles. The had chosen 63372 articles with 47283 articles to serve as the training data and the remaining 16089 articles were used for testing articles with all articles classified into 5 categories and 80 sub-categories:

The process of classification can be divided into 2 major parts: One part is to establish the classification models from the already classified data; the other is to use the classification models to categorize from the unclassified data [1, 2, 4, 6, 7, 8].

We refer to the dissertation of K Aas and L. Eikivil[1] and sum up the systemic process for document classification as Fig. 2.

At first, we divide the article required for classification into training data and testing data. From the training data of already classified category, we pick out the similar feature terms or key words. This process is known as Feature Extraction including preprocessing, document representation and feature selection, totally 3 parts.

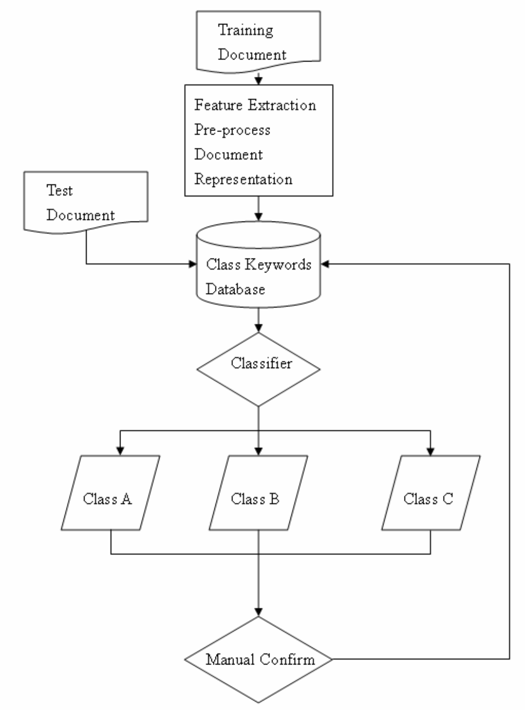

Fig. 2 Document Classification Process

The feature extraction is mainly aimed to extract the representative features from document sets. Typically, we surmise that the word of higher occurrence frequency within articles will be of more importance. However, actually, it shows no highly positive correlation between words of high occurrence frequency and context importance.

The document representation means that a document is converted into the format available for computer processing. Typically, they are denoted as below vector modes: (W_1, $T_{1;}$ W_2, $T_{2;}$ …; Wn, T_n) wherein W_1 means weighting values and T_i means feature terms (i=1…n). The weighting calculation is oftentimes adopted with Boolean Weighting, Word Frequency Weighting, TFIDF Weighting, and Entropy Weighting [1, 9, 10, 11].

SYSTEM ARCHITECTUR

Within this article, it is based on the "term database comparison" within the text classification and we propose another additional category priority and allow for the improved classification algorithm for multi-classification so that we can improve the accuracy of text classification.

Class Priority

As we mention in chapter 1, when the key word differ mutually, we can give the higher priority to the longer key words because the longer term will come with more edges the shorter ones. If the word numbers of terms are the same, we hereby propose a notion of "class priority". Fro example, the food category generally poses higher edge over other material categories; thus, the food is allowable for higher priority.

To sum up, when the word numbers of key words are different, it exists in no priority problem. Simply, we can refer to the longer key words for basis; if the word numbers of key words are all the same, we can refer to the class priority for basis thereafter.

However, within the process of dynamic priority rule input, this process can be known as a cycle, namely redundant priority. For example, during the leaning process, we find that both "wood" and "house" oftentimes occur simultaneously with the general meaning of "wooden furniture". Thus, we can add a priority rule: 94 (furniture) → 44 (wood category). Also, we find that the wooden dog house actually never belongs to household commodity but it is classified into wooden commodity. If based on this finding, we can add the priority rule 44→94 and it results in the cycle priority.

In general, the notion of priority is traversable. For example, A→B→C means that this method is available for the Direct Acyclic Graph (DAG) to represent the current priority sequence.

If there were 3 categories of A, B and C, A is prior to B and it can be denoted as (A→B). Whenever B is superior to C, it represents (B→C). From the traversable rule, we can derive the result that A is prior to C. If we decelerate C is prior to A that declaration will form a cycle priority (A→B→C→A).

The above graph shows that the unallowable cycle priority is known as the redundant priority. Also, it can be denoted as (Redundant Class → A≡B≡C).

We know that under the situation of NP-complete, even if n is extremely small, the time complexity still grows rapidly. However, if we want to find a cycle, the complexity can be reduced to O(n). "n" means the category numbers from the levels. When A≡B≡C happening, actually, we can combine them as a category by re-defining and re-naming. According to this feature, we can find out the cycle one by one and then delete the found cycle. All the redundant priorities will be deleted and the processing notion can be briefed as below:

The 3 steps to delete the redundant cycle:

1. Determine the initial category (Specify a certain category as the root class.).
2. By means of "Deep First", we can move to each sub-category of the initial category and check if there is any cycle priority formed. If so, delete the cycle priority.
3. If there is more than 1 root class, we can view another root class as the initial category to execute the step 2 till all the categories have been processed to end up the process.

The ultimate goal of above process is aimed to obtain a category level of no cycle priority. To put it differently, we can convert the category level graph into the Direct Acyclic Graph.

The complete method to delete the cycle priority algorithm start with the root class firstly by means of “deep firstly” searching. We can search each category and sub-root-class v (denoted as descendant (v)). Thereafter, we can check the route of this node to see if it exists in the category W to make the cycle priority {W,…V,W}. Thereafter, we can further delete this cycle. The whole set can be re-defined as a new category W’; namely, we can re-define a new category to replace the category set of cycle priority shown as Fig 3.

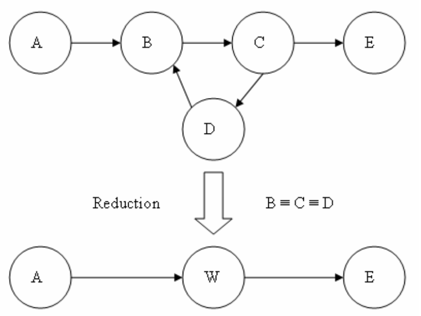

Fig. 3 The Demonstration for the Combination of Redundant Priority

Algorithm CPD (V) (Cyclic Priority Detection)
Input: There is a Direct Graph G = (V, E) recorded with all priority correlations.
Output: There is a Direct Acyclic Graph without any cycle priority existing, neither does the redundant category.

Pre-processing: We can firstly find out the descendant (v) from all the sets of category {V}.

```
1) visited [v]= true
2) for (each class w adjacent to v) begin
      (a) we attached next being traversed class w to the
          descendant (v) set.
      (b) Check the descendant (v) whether there is repeated
          class w appeared. If so, we reduce these redundant
          classes {w,…v,w} into another class w’ and
          reconstruct the graph with the class w’; otherwise
          we can discard the class w from the descendant (v).
      (c) If cyclic priority is found then   CPD(w’)   Else
          CPD(w)   end
```

Multi- Classification

Knowing from the previous studies, some key words like “container ship”, occur will frequently cause erroneous classification because there are more than 2 category of commodity within the cargo. Thus, this research article firstly filters out the data with such a feature and then it is particularly processed. Thus, it can considerably improve the accuracy of subsequent classification.

The major basal accordance for filtering process is that after each learning cycle, from the erroneous data, we can pick out the key words highly possible to cause wrong multi-classification. Before the action for general classification, we previously remove all the data featured with multi-classification. In other words, we select all the erroneous data from the original multi-classification; and then, after processing, the data will be correctly classified.

System Architecture

From the sub-category filed of CCC Code, we can extract the original term pools; by following experimental rules, we can add the pre-defined key words with priority weight values and multi-classification. Thereafter, we can further add training data to execute the first classification. Thus, we can use the priority rules proposed by this research and the revised key word to execute learning; finally, we can examine the new term pool and rules.

Within the pre-processing, we can execute 3 actions:

1. Delete the stop words like conjunctions, prepositions and article nouns.
2. Delete the particular characters like “@” and “#”.
3. Process the syntactic arrangement. Reduce the gerunds, plurals, past tenses, present participle and adverbial ending to their original forms.

Under multi-classification processing, we have filtered out some key words with multi-classification before the 1st classification, such as “container ship”. We can filter these key words of B/L from the training data independently. The data is allowable for direct multi-classification. According to the classification situation, we can gradually revise or add the key words with such a feature.

Regarding the major classification processing, we can directly adopt the key words by means of Fully Match and Partially Match to compare with training data under the 1st classification. The so-call Fully Match Methodology defines the successful action that the check-out key word shall be fully matched to reach its successful comparison. The Partially Match Methodology means the successful action that the prefix of check-out key word shall be fully matched to reach its successful comparison. When we are checking out the key words, the longer words shall pose higher priority such as the situation that the “ice cream” shall be prior to the “ice”.

In addition, we can set up an information table for key word priority and place this table into the same location as the key word database. The priority default for each category is set to 1. As for the priority determination, we can default in the notion that the finished product category shall be prior to the raw material category according to our past experience. For example, the priority of “Furniture” shall be superior to “Wood”. “Furniture” belongs to sub-category 94, while “Wood” falls on sub-category 44. Thus, we can add the category priority rule (94→44) and the priority weighing value of sub-category 94 will be added by 1. Whenever there is any new priority rule added, we can use the foresaid CPD Algorithm to check if it exists in the redundant cycle priority.

After various steps of classification, we can collect the data unavailable for classification and the data of erroneous classification to serve as the learning reference.

Finally, when we are executing of leaning steps, we have arranged the sequence for 97 sub-categories according to the previous classification results. It also means that we can check out for modification to give higher priority of category. Also, after dynamically adding each new subsequence, we can use CPD Algorithm to examine whether it will cause the redundant cycles; if so, delete them. The learning process mentioned in this research is aimed to correct the "information table of category priority", "key words" and "multi-classification key words" , totally 3 parts and we can also use the testing data to examine if the systemic classification accuracy after correction is well improved.

EXPERIMENTATION

The data resources of this research are adopted from the export B/L with the major classification fields like the commodity description fields. We take 100,000 B/L records to serve as the training data for the 1st classification. Following that, according to the classification results, we con revise the term pools and priorities. After completeness, we further evaluate 4000 records of testing data. The classification is executed based on CCC Code with the commodity features categorized as 21 categories and 97 sub-categories. The classification of this research is operated mainly based on 97 subcategories.

Results of Training Data

The Records of Training Data	100000 records
Key Words	10664 words
Category Priority Rule	No
Multi-classification Key Words	No
Congruent to the key word category with accurate number of records	39519
Congruent to the key word category with wrong number of records	17005
Incongruent to the key word category yet with accurate number of records	423
The entirely unclassified records	43053
Average accurate rate	63%
Average recycle rate	60%

Fig. 4 The 1st Classification Results

When we are executing the 1st classification process, we totally check out 10664 key words. To examine about the extent that the added subsequences and the filtered words after multi-classification will affect the systemic accuracy, we do not determine the category priority and pause the filtering for key words of multi-classification, we compare from the increasing coding number order one by one. Because there is no priority existing, if the key words repeat matching, this B/L will be viewed as the classification errors. After finishing the classification, we add supplementary artificial counting to reach accuracy and recycle rate. The result is shown as Fig. 4. We can exclude the 43053 records unavailable for classification with only 63% of accuracy rate reached. This is because the data is extracted from CCC Code yet not from training data.

Regarding the last process, we add 17 sets of category priority rules into category system and 14 key words of multi-classification. Among them, we filter out the key words featured with multi-classification from B/L contexts independently for another step of processing. When the said data is matched with various key words, we still view kit as the accurate classification. If the remaining data is matched repeatedly but featured with various priorities, we therefore refer to the key words with higher priorities shown as Fig. 5. The average accuracy rate is improved to 96% and it is well proven that adding accurate priorities and allowance for multi-classification will remarkably assure of excellent classification accuracy.

The Total Records	100000 records
Key Words	12006 words
Category Priority Rule	17 sets
Multi-classification Key Words	14 sets
Congruent to the key word category with accurate number of records	70515
Congruent to the key word category with wrong number of records	2939
Incongruent to the key word category yet with accurate number of records	13
The entirely unclassified records	26533
Average accurate rate	96%
Average recycle rate	92%

Fig. 5 The Classification Results After Using the Revised Term Pools with Multi-classification Added

Results of Testing Data

The Total Records	40000 records
Key Words	12006 words
Category Priority Rule	17 sets
Multi-classification Key Words	14 sets
Congruent to the key word category with accurate number of records	10243
Congruent to the key word category with wrong number of records	535
Incongruent to the key word category yet with accurate number of records	5
The record number of completely unclassified records	29217
Average accurate rate	96%
Average recycle rate	91%

Fig. 6 The Classification Results After Using the Revised Term Pools with Multi-classification and Category Priority Added

Shown as Fig. 6, knowing from the experimental data of this chapter, whenever the data resource is adopted from some fixed ranges such as the B/L contexts of this research with the major fields listed with English words, after we execute the learning correction by checking out the key words, we can definitely reach 80% accuracy rate. Plus the addition of category priority rules and allowance for multi-classification, we can improve the accuracy rate up to 95% above.

CONCLUSIONS

This research article is adopted with the Free Content Methodology to convert the stored data into the format available for computer processing. Also, we attempt to use the notion “allowance for multi-classification” and “deletion of redundant cycle priority” applicable to the sequence arrangement of category priority. The experimental results show that the accuracy rate before usage of foresaid 2 notions only reaches 78%, but the ultimate accuracy rate can be improved to 96% after adding the rules with the recycle rate kept at 91%. Thus, the major contribution of this research article is aimed for (1) the completeness of the determination for “allowance of multi-classification”; (2) the proposal of “detection for redundant cycle priority algorithm” with the application for the design of category priority. It is experimentally proven that it can effectively improve the classification accuracy rate for the commodity fields with English contents listed.

This research article combines the redundant cycle categories into a large category. In the future, we can further research the weighing values for various categories so that we can identify what the exact category to belong.

When the situation of redundant cycle category does not cover the overlapped parts of category, the cycle of (A→B→C→A) can divided into (A-C)→B, (B-A)→C, (C-B)→A with the cycle separated availably.

REFERENCES

[1] K. Aas and L. Eikvil,” Text categorization: A survey,” Technical report, Norwegian Computing Center, 1999.

[2] Chris D. Paice, “Another stemmer,” ACM SIGIR Forum, vol. 24, pp. 56-61, 1990.

[3] K.A. Hamill and A. Zamora, “ The use of titles for automatic document classification,” Journal of American Society for Information Science, vol. 31, no. 6, pp. 386-402, 1980.

[4] D.D. Lewis, R.E. Schapire, J.P.Callan, and R. Papka, “Training algorithms for linear text classifiers,” Proceedings of SIGIR-96, 19th ACM International Conference on Research and Development in Information Retrieval, pp. 298-306, 1996.

[5] M.E. Maron, “Automatic indexing: an experimental inquiry,” Journal of the ACM, pp. 407-417, 1961.

[6] M.E. Porter, An algorithm for suffix stripping,” Program, vol. 14, no. 3, pp. 130-137, 1980.

[7] Michael. J.A. Berry and Gordon S. Linoff, “Data mining Techniques: for marketing, sales, and customer support,” USA, John Wiley & Sons, Inc, 1997.

[8] Michael. J.A. Berry and Murray Browne, “Understanding Search Engines: Mathematical Modeling and Text Retrieval,” USA, 1999.

[9] O.W. Kwon and J.H. Lee, “Web page classification based on k-nearest neighbor approach,” Proceedings of the 5th International Workshop on Information Retrieval with Asian Lanfuages, pp. 9-15, 2000.

[10] F. Sebastiani, “Machine learning in automated text categorization,” ACM Computing Surveys, vol. 34, 2002.

[11] Yiming Yang and Jan O. Pedersen, “A Comparative study on feature selection in text categorization,” Proceedings of {ICML}-97, 14th International Conference on Machine Learning, pp. 412-420, 1997.