BRF: A Framework of Retrieving Brand Names of Products in Auction Sites

Ivy Hong-Ieng Hoi, Mars Liwen Liao, Chih-Chieh Hung, and Evans Tseng

EC-Central Engineering Team, Yahoo! Taiwan
{crazygirl8170,smalloshin}@gmail.com,
{lliwen,tobala}@yahoo-inc.com

Abstract. Online auction sites give sellers extreme high degree of freedom to fill in the product information so that they can promote their products to attract bidders in many ways. One of the most popular ways to promote is to add brand names and model names in their product titles. However, the side effect of this promotion way is that the search results are seriously irrelevant to what users expect, especially when brand names are used as query terms. In this paper, we target at the problem of retrieving the brand name of a product from its title. First, the root causes could be categorized into three types by observing the real data on the online auction site of Yahoo! Taiwan. Then, a brand-retrieving framework BRF is proposed. Specifically, BRF first eliminates those brand and model names, which may not be the actual brand name of this product, in a product title; then BRF represents a product title by selecting representative keywords with their importance; finally, BRF models the problem as a classification problem which identify what the brand name (class) of a product title is. Extensive experiments are then conducted by using real datasets, and the experimental results showed the effectiveness of BRF. To best of our knowledge, this is the first paper to design a mechanism of retrieving the brand names of products in auction sites.

Keywords: e-commerce, brand name, auction, information retrieval.

1 Introduction

Without taking valuable time to search for an item from store to store, e-commerce provides users with a more convenient way in which to shop. Among several models in e-commerce, online auction is one of the most popular and effective ways of trading by participants bidding for products and services over the Internet. Online auction sites[1], such as Ebay in US, Taobao in China, and Yahoo! Auction in Taiwan, are perfect examples of this business model. One of the main characteristics why auction sites are so popular is the freedom. That is, an auction site provides a platform which allows users to fill out any information of their products, including the title, description, price, and so on. With such freedom, sellers on an auction site can promote their products to attract bidders in many ways.

[1] Online auction site is abbreviated as auction sites in the following sections of this paper.

C. Huemer and P. Lops (Eds.): EC-Web 2013, LNBIP 152, pp. 1–13, 2013.

Table 1. Statistics of how users find out products (2013/3/11~2013/3/17)

E-commerce properties in Taiwan	Category Navigation	Search by Queries
Shopping Center(B2C)	47.04%	5.54%
Shopping Mall(B2B2C)	12%	6.37%
Auction (C2C)	1.90%	17.40%

One of the most popular ways to promote is to add brand names and model names in their product titles. This approach can help sellers promote their products effectively since the proportion of the queries including brand names always exceeds 10% in top-1000 queries, according to the report of Yahoo! Auction Taiwan. However, you have got to take the good with the bad. The side effect of using this promotion way causes the search results irrelevant to what users expect when brand names are used as query terms. Such irrelevancy is seriously harmful to user experience in search since searching plays a much more important role in auction sites than other business models. Table 1 shows the statistics how users find out the products they want. It can be seen that users highly depends on search in auction sites. To improve the relevancy of the search results, it is a crucial issue to retrieve the actual brand names of products. Once the actual brand names of products can be retrieved, search engines in auction sites could extract the products with the given brand names precisely.

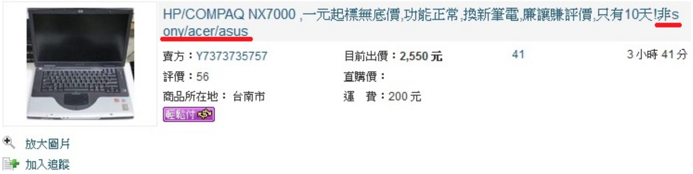
In this paper, we target at the problem of retrieving brand names of products in auction sites. A naïve method to find the actual brand names is to use a dictionary which contains a set of brand names and the mapping from model names to brand names. If there is any matched brand or model names in the product title, this match brand name can be identified as the actual brand name of this product. However, this naïve approach may fail since the brand names and model names are usually abused to promote products, the product titles may contain noisy and irrelevant information. Therefore, this paper proposed a framework, called BRF (standing for Brand-Retrieving Framework) to find the actual brand names of the products in auction sites. To best of our knowledge, this is the first paper to design a mechanism to solve this problem. Several issues remain to be addressed to effectively retrieve the actual brand names from product titles:

1. False-Use Problem

To improve the exposure rates of their products, sellers may add famous brand names or model names in their product titles. Buyers may obtain irrelevant results with respect to their queries. For example, Fig. 1 shows two products of the search results for query Acer in the auction site in Yahoo! Taiwan. The first one is relevant to the query Acer since the product is Acer 4820TG whereas the second one is irrelevant to the query Acer since this seller is going to sell his lab-top HP/COMPAQ NX7000. The reason is that the title contains the other famous brand names (i.e., sony, acer, and asus) which are underlined in the product title[2]. To address this issue, BRF uses the

[2] 非 in Chinese means “non”.

brand-model map to eliminate those keywords in the product title which may interfere the identification of the actual brand names of products.


HP/COMPAQ NX7000 ,一元起標無底價,功能正常,換新筆電,廉讓賺評價,只有10天!非sony/acer/asus

賣方：Y7373735757 目前出價：2,550 元 41 3 小時 41 分

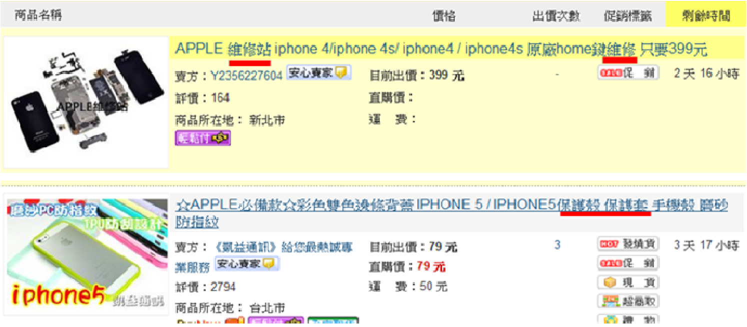
評價：56 直購價：

商品所在地： 台南市 運　費：200 元

Fig. 1. An example for False-Use Problem

2. Accessory/Repair-Related Problem

Some accessories and maintenance service providers add the brand names into the titles of their products as well. The reason is that they intend to specify the brand names that their accessories can fit or their maintenance services can support. However, when users submit brand names as queries, they are likely to find the products instead of accessories [3]. For example, Fig. 2 shows the search results of query “apple”. It can be seen that the product on the top is a protective shell for iPhone, which title contains “apple”. The search results of accessories and maintenance services may disrupt users to find the products with the specific brand names due to the fact that the amount of the accessories and maintenance services are usually much larger than that of the products which specific brand names, especially for the famous brand names[3]. To address this issue, BRF uses the feature generation mechanism, which is based on Chi-Square and TF-IDF, to detect the features that represents for accessory and maintenance. Thus, these accessories and maintenance services can be distinguished from the actual products with the specific brand names.


商品名稱 價格 出價次數 促銷標籤 剩餘時間

APPLE 維修站 iphone 4/iphone 4s/ iphone4 / iphone4s 原廠home鍵維修 只要399元

賣方：Y2356227604 安心賣家 目前出價：399 元 - 促銷 2 天 16 小時

評價：164 直購價：

商品所在地： 新北市 運　費：

☆APPLE必備款☆彩色雙色邊條背蓋 IPHONE 5 / IPHONE5保護殼 保護套 手機殼 磨砂防指紋

賣方：《凱益通訊》給您最熱誠專業服務 安心賣家 目前出價：79 元 3 發燒貨 3 天 17 小時

直購價：79 元 促銷

評價：2794 運　費：50 元 現貨

商品所在地： 台北市 超商取

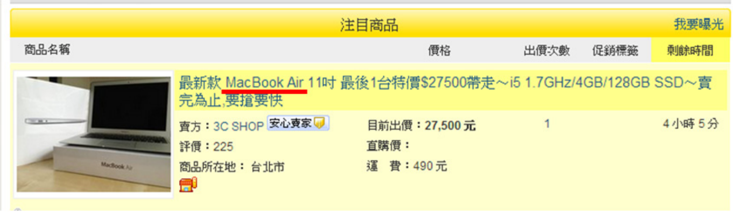
Fig. 2. An example for Accessory/Repair-Related Problem

3. Brand-Name-Omission Problem

Brand-Name-Omission Problem indicates that sellers may use model names to describe their products without mentioning the brand names. This problem usually

[3] 維修 in Chinese means “maintenance”. 保護殼, 保護套 in Chinese means “protective shell”.

happens in those products with the famous brand names. The reason is that people usually call these popular products by their model names instead of mentioning their brand names in their daily life. Fig. 3 shows an example that there is only the model name “MacBook Air” in the product title. However, the actual brand name “Apple” is missing. Therefore, this product cannot be found if sellers use “apple” as the query term to search.


注目商品 我要曝光

商品名稱 價格 出價次數 促銷標籤 剩餘時間

最新款 MacBook Air 11吋 最後1台特價$27500帶走～i5 1.7GHz/4GB/128GB SSD～賣完為止,要搶要快

賣方：3C SHOP 安心賣家 目前出價：27,500 元 1 4 小時 5 分

評價：225 直購價：

商品所在地： 台北市 運 費：490 元

Fig. 3. An example for Brand-Name-Omission Problem

To address this problem, BRF proposed the feature generation mechanism so that the model names can be extracted, and the classifier can use the labels from training data to map the specific model names to their brand names.

The rest of this paper is organized as follows. Section 2 presents some statistics to show the impact of the three issues. Section 3 then describes the brand-retrieving mechanism BRF for finding brand names from product titles. Section 4 presents experimental results. Finally, Section 5 draws conclusions

2 Preliminary

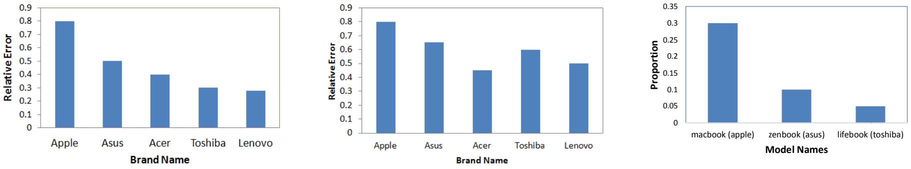
This section conducts some preliminary experiments to show the impacts of three problems: False-Use Problem, Accessory/Repair-Related Problem, and Model-Convention Problem. In the following experiments, the dataset is obtained the auction site of Yahoo! Taiwan, including the query logs and the search result pages. The target category is the laptop category. The duration is from December 3rd to December 6th, 2013, totally 3 days. Since the search result pages may vary anytime, we sample search result pages for three times every day.

In general, these three problems may reduce the relevancy of the search results of queries. To evaluate the relevancy of the search results of queries, Query Click through Rate (QCTR) is one of the important metrics where QCTR is defined as the number that the query search results clicked divided by the number of queries reported. For example, if there are 200 search results clicked out of 1000 queries, the QCTR would be 20%. Obviously, the higher the QCTR is, the more relevant the search results of queries are.

2.1 False-Use Problem

The False-Use Problem leads to the decrease of QCTR. To show the impact of this problem, a preliminary experiment is conducted. The queries are the brand names.

For each query, we compute the QCTR of top-100 results extracted from two sources: (a) the original search results and (b) the refined search results which are exactly the products with brand names. Then, the relative error of QCTR of different brand names is computed. Obviously, the larger the relative error of QCTR is, the larger impact of False-Use Problem is. Fig. 4(a) shows the experimental results. It can be seen that “apple” has the largest QCTR difference, showing that many products in top-100 results are not clicked by users although their titles contain the brand name apple. This experiment shows that the False-Use Problem significantly reduces QCTR.



(a) (b) (c)

Fig. 4. Impact of (a) False-Use Problem, (b) Accessory/Repair-Related Problem, and (c) Brand-Name-Omission Problem

2.2 Accessory/Repair-Related Problem

Accessory/Repair-Related Problem indicates that the search results contain not only the products but also a list of accessories and maintenance services containing brand names when queries are brand names. To show the impact of this problem, a preliminary experiment is conducted. Following the same setting and dataset as the previous experiment, Fig. 4(b) shows the relative error of QCTR. It can be seen that the relative error of QCTR of each query term is larger than 0.5. Specifically, the relative error of QCTR of “Apple” is almost 0.8 while there are around 85% accessories and maintenance services in the top-100 search results in average. This result supports the claim that customers tends to find the products, instead of accessories and maintenance services, when they submit brand names as queries.

2.3 Brand-Name-Omission Problem

The Brand-Name-Omission problem indicates that sellers may use model names to describe their products without mentioning the brand names. Consequently, their listings will not be viewed by potential buyers who use brand names as their query terms. This problem usually happens in those products with the famous brand names. The impact of this problem is the potential revenue loss for both sellers and auction sites: since users may not use the convention terms to find the desired products, sellers are hard to sell their products and auction sites cannot get the commission charges from sellers. For example, users interested in iPhone 5 all know that it is a cell phone by Apple, and therefore might expect to see all listings of iPhone 5 for sale when a query term 'apple' is used. However, a listing of this model without the actual word 'Apple' in the title will not appear in the search results. To investigate the impact of this problem, a preliminary experiment is designed. We submit each selected model name as a query, and two types of the products can be obtained: a) listing titles with model names only and b) listing titles with both model and brand names. Let the numbers of the former and the latter types be Na and Nb. Then the proportion value is computed. Obviously, the larger the proportion is, the more products are likely to be found hardly if the queries are brand names. Fig. 4(c) shows the results. It can be seen that the query “macbook” has the largest proportion, which means many sellers usually use “macbook” only in the listing titles without specifying the brand “apple”. This case shows that around 30% macbook cannot be found in the search results of “apple”, which may cause both sellers’ and auction sites’ loss of revenue.

3 BRF: Brand-Retrieving Framework

3.1 System Architecture

To retrieve the brand names of products in auction sites, BRF models this problem as a classification problem which identify the brand name by given the product title. BRF is composed of two stages: training stage and classifying stage. In the training stage, the training dataset is given to train the classifier to generate the model for classification, where each entry in the training dataset is a pair of the product title and its brand name. The training stage is composed of three phases: 1. preprocessing, 2. feature generation, and 3. classifier training. In the preprocessing phase, notations and terms which are not helpful for classification will be eliminated. In the feature generation phase, terms with highly related to brand names will be extracted, say representative keywords, and each product title will be represented as a vector of representative keywords. In the classifier training stage, every pairs of (vector, brand) is used to train the classifier to generate the model. After generating the model, in the classifying stage, the product title will be presented into a vector of representative keywords generating in the training stage. Given the vector, the brand name of the product title can be predicted by the model. The technical details will be described in the later sections.

3.2 Preprocessing

Given the product title as the input, the main goal of this phase is to filter out the words that may decrease the accuracy of classification.

First of all, the conventional preprocessing approaches should be adopted, such as tokenization, punctuation elimination, stop word elimination, and so on. Besides these, we should also handle the False-Use Problem in this phase. As mentioned above, the False-Use Problem is that there are the other brand names in the product title. Thus, it is necessary to eliminate these irrelevant brand names in the product title so that our system can extract the actual brand name of this product more accurately. To eliminate the irrelevant brand and model names, the brand-model map is defined as follows:

Definition. Brand-Model Map
Let the product title be a sequence of tokenized words <s_1,s_2,s_3,...,s_n>. The brand-model map is a sequence <b_1,b_2,...,b_n>. Each entry b_i is the number of brand or model names in <si-δ,...,si+δ> where δ is a specified parameter if s_i is a brand or model name. Otherwise, the value of b_i is 0.

For example, let δ be 2. Given a product title <Apple MacBook 13.3" Laptop, non Samsung, HP, Asus, Acer, Lenovo>, the brand-model map can be derived as <2, 2, 0, 0, 0, 3, 4, 5, 5, 4>.

Obviously, each entry in the brand-model map represents how dense the brand and model names appear surrounding a brand or a model name. According to the brand-model map, the product title <s_1,s_2,…,s_n> can be partitioned into subsequences <σ_1, σ_2, …, σ_m> such that the total variance of these subsequences, i.e., $\sum_{i=1}^{m} \mathrm{Var}(\sigma_i)$, is minimized. The variance of a subsequence is $\mathrm{Var}(\sigma_i) = \sum_{b_j \in \sigma_i} |b_j - \mu| / |\sigma_i|$, where μ is the mean of all values in σ_i. To achieve this goal, we can borrow algorithm TC in [1]. The average of each subsequence is then computed. Given a parameter ε, those subsequences with average values larger than ε will be eliminated.

Following the example above, since the brand-model map is <2, 2, 0, 0, 0, 3, 4, 5, 5, 4>, we can partition the product title as <σ_1:(Apple MacBook), σ_2: (13.3" Laptop, non), σ_3: (Samsung, HP, Asus, Acer, Lenovo)>. Let ε be 4. Since the average value of σ_3 is (3+4+5+5+4)/5 = 4, the subsequence σ_3 is then eliminated from the product title.

The value of δ and ε are determined by the behavior how users add irrelevant brand and model names into the product title. In our current cases, the value of δ and ε can be set as 2 and 4 to achieve the acceptable results (specifically, the precision of classification is around 70%). The setting of δ and ε could be also trained by machine learning approaches. This issue is beyond our scope here and left as the future work.

3.3 Feature Generation

This section describes how to represent a product title into a vector. This phase consists of two steps: 1. Selecting the representative keywords, and 2. Determining the importance of representative keywords.

3.3.1 Selecting the Representative Keywords

Given the candidate keywords obtained from the previous phase, say {k_1,k_2,…,k_m}, in this phase, we are going to select the most representative keywords by Chi-Square attribute evaluation (χ^2). Note that the format of training data will be a pair of a product title (represented by a set of candidate keywords) and its brand name c_j.

Generally speaking, Chi-Square attribute evaluation evaluates the worth of a feature by computing the value of the chi-squared statistic with respect to the class [2]. To compute the value of χ^2 for multiple classes (i.e., brand names), we start from the definition of the Chi-Square value of a term k_i with respect to a class c_j: $\chi^2(k_i, c_j)$. Let A be the number of the term k_i belonging to the class c_j, B be the number of the term k_i not belonging to the class c_j, C be the number of the terms which are not k_i but belong to class c_j, and D be the number of the terms which are not k_i and not belonging to the class c_j. Supposing N=A+B+C+D, $\chi^2(k_i$,c) can be defined as follows:

$$\chi^2(\mathrm{k_i}, \mathrm{c_j}) = \frac{\mathrm{N} \times (\mathrm{AD} - \mathrm{CD})^2}{(\mathrm{A} + \mathrm{C}) \times (\mathrm{B} + \mathrm{D}) \times (\mathrm{A} + \mathrm{B}) \times (\mathrm{C} + \mathrm{D})}$$

Based on $\chi^2(k_i, c_j)$, we could require to discriminate well across all classes. Then we could compute the expected value of for each candidate keyword as follows:

$$\chi^2_{\mathrm{avg}}(\mathrm{k_i}) = \sum_{\mathrm{j}} \mathrm{Pr}\ (\mathrm{k_i}) \times \chi^2(\mathrm{k_i}, \mathrm{c_j})$$

After computing the expected values for each candidate keyword, the candidate keywords with top-n χ^2_{avg} value are selected to be the representative keywords, say $\{t_1, t_2, \ldots, t_n\}$.

3.3.2 Determining the Importance of Representative Keywords

In this step, TF-IDF is used to determine the importance of representative keywords. TF-IDF is the product of two statistics, *term frequency* and *inverse document frequency*. For ease of presentation, t represents a representative keyword, d represents a product title, and D represents the set of product titles.

For the term frequency TF(t,d), the simplest choice is to simply use the raw frequency of a keyword in a product title. However, to prevent a bias towards longer product title, the normalized frequency is used. Let the raw frequency of t be f(t,d), the normalized frequency is defined as follows:

$$\mathrm{TF}(\mathrm{t}, \mathrm{d}) = \frac{\mathrm{f}(\mathrm{t}, \mathrm{d})}{\max\{\mathrm{f}(\mathrm{w}, \mathrm{d}) | \mathrm{w} \in \mathrm{d}\}}$$

The inverse document frequency IDF(t,D) is a measure of whether the representative keyword t is common or rare across the set of product titles D. It is obtained by dividing the total number of product titles by the number of product titles containing the representative keyword, and then taking the logarithm of that quotient. Formally, the inverse document frequency IDF(t,D) can be defined as follows:

$$\mathrm{IDF}(\mathrm{t}, \mathrm{D}) = \log \frac{|\mathrm{D}|}{|\{\mathrm{d} \in \mathrm{D} | \mathrm{t} \in \mathrm{d}\}}$$

Finally, the value of TF-IDF is defined as follows:

$$\mathrm{TFIDF}(\mathrm{t}, \mathrm{d}, \mathrm{D}) = \mathrm{TF}(\mathrm{t}, \mathrm{d}) \times \mathrm{IDF}(\mathrm{t}, \mathrm{D})$$

A high weight in TF–IDF is reached by a high term frequency in the given product title and a low document frequency of the representative keyword in the whole collection of product titles. As a representative keywords appears in more product titles, the ratio inside the logarithm approaches 1, bringing the idf and tf-idf closer to 0.

To put it together, a product title *d* can be represented as a vector <TFIDF(t_1,d,D), TFIDF(t_2,d,D),…, TFIDF(t_n,d,D)> where t_i is the *i*-th representative keyword.

3.4 Classification Strategy

This section describes a possible classification strategy to use the features generated by the previous step. Here, we do not claim that this strategy is the best among all others, but it is just a reasonable strategy. Our feature generation methodology has good usability for the following reasons. First, it is neutral to classifiers. That is, it can be used for any classifiers, including decision trees, rule-based classifiers, and support vector machines. Second, it can team up with other feature generation methodologies. Products may have multiple attributes, including prices, description, category information, and so on. Some of these attributes may be able to transform into as a single value, an interval, or a sequence of values per product. Handling numerical attributes is beyond the scope of this paper. In our study, three classifiers are built using the naïve Bayesian classifier, decision tree, and support vector machine (SVM) [4]. According to the experimental results and the characteristic of e-commerce, SVM is implemented in our current system. This design decision stems from two characteristics of the feature vectors generated. First, they are high-dimensional since many features may be generated. Second, they are sparse since each product title has only a few of these features. The SVM is well-suited for such high-dimensional and sparse data [5].

4 Experimental Results

4.1 Environment Settings

In the following experiments, three datasets are used: Shopping, Auction(small), and Auction(big). These three datasets are used to test the performance of the proposed mechanism in different conditions. In these three datasets, the products in the *laptop* category during 2013/3/11 to 2013/3/17 are extracted to evaluate the proposed mechanism. The characteristics and the statistic information of each dataset are provided as follows:

Shopping dataset contains the products of Shopping Center in Yahoo! Taiwan which is a B2C platform. The suppliers describe the characteristics of laptop clearly in the product title. The product title usually contains at least the brand name, the model name, and specification of the laptop. Thus, we can observe that the product titles in this dataset precisely describe information about the laptop without noise. Moreover, almost all the products under the laptop category are actually products of laptops. In this dataset, we totally collect 2906 products with ten major brands of laptop. For each brand, there are at least 50 products. The number of products of each class is listed in Table 2.

Table 2. Statistics of Shopping Datasets (2013/3/11~2013/3/17)

Brand Name	Number of Products	Brand Name	Number of Products
Acer	904	HP	297
Apple	94	Lenovo	210
Asus	516	MSI	78
Dell	153	Sony	95
Fujitsu	83	Toshiba	476

Auction(small) and *Auction(big)* are the datasets from Yahoo! Auction, Taiwan. Auction(small) provides an ideal case that the number of products of each brand is uniformly distributed whereas Auction(big) provides the real situation that the number of products of each brand exists some bias. In both datasets, most of the product titles do not describe the characteristics of laptops very clearly. To increase the exposure rate, the sellers may describe unrelated information in the title. Besides laptops, there are many irrelative products which are related to maintenance service, notebook accessories products and the remaining unrelated products are probably in other categories. We classify products into 15 classes where 12 classes are brands of laptops which each brand also contains at least 50 products. The remaining three classes are “maintenance”, “accessories” and “others” which refer to maintenance service providers, accessories and products not in the above classes respectively. The number of products for each brand is shown in Table 3.

Table 3. Statistics of Auction(small) and Auction(big) Datasets (2013/3/11~2013/3/17)

Brand Name	Auction(small) Number of Products	Auction(big) Number of Products
Acer	50	54
Apple	50	132
Asus	50	169
Clevo	50	50
Dell	50	153
Fujitsu	50	126
Gigabyte	49	49
HP	50	135
Lenovo	50	171
MSI	50	189
Sony	50	136
Toshiba	50	202
Other Brands	50	74
Maintenance	50	295
Accessories	50	925

In this experiment, three classifiers are used: Naïve Bayes, J48, and SVM. These classifiers are from the WEKA library []. *Precision* is the metric used to evaluate the performance of these classifiers under different datasets. Formally, the precision of a classifier is defined as $\frac{1}{n}\sum_{k=1}^{n}\frac{N_{k,c}}{N_k}$, where N_k is the number of the products in class k and $N_{k,c}$ is the number of the products that are in class k and successfully classified as class k. Ten-fold cross validation is used to evaluate the effectiveness of each classifier in the proposed mechanism. Specifically, the dataset will be randomly partitioned into ten complementary subsets of near equal size, and one of the subsets will be assigned as the testing data while the remaining are used as the training data.

4.2 Precision in Three Datasets

In this section, we conduct the experiments that test the precision of three classifiers in three datasets.

Fig. 5 shows the results of shopping dataset. It can be seen that the precision of each classifier exceeds 97%. In addition, J48 and SVM achieve the best performance that precisions of these two classifiers are over 99%. The reason is that the product title of each product in this dataset not only points out product characteristics clearly but also has not any interfering information. Thus, this result shows that the proposed method can work well when the given product titles are of less noise.

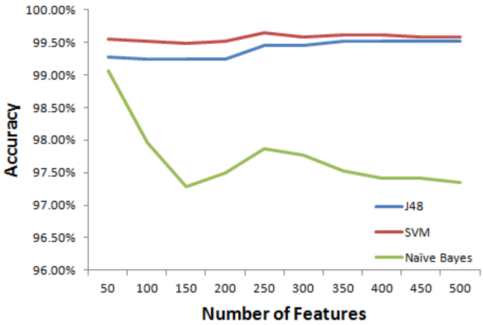
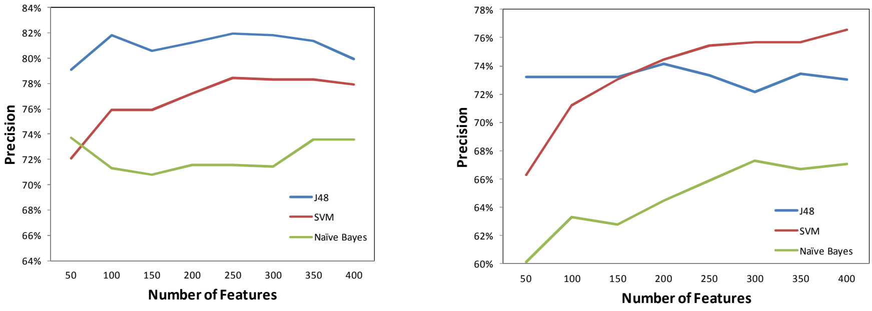


Fig. 5. Precision with different classifiers: Shopping Dataset

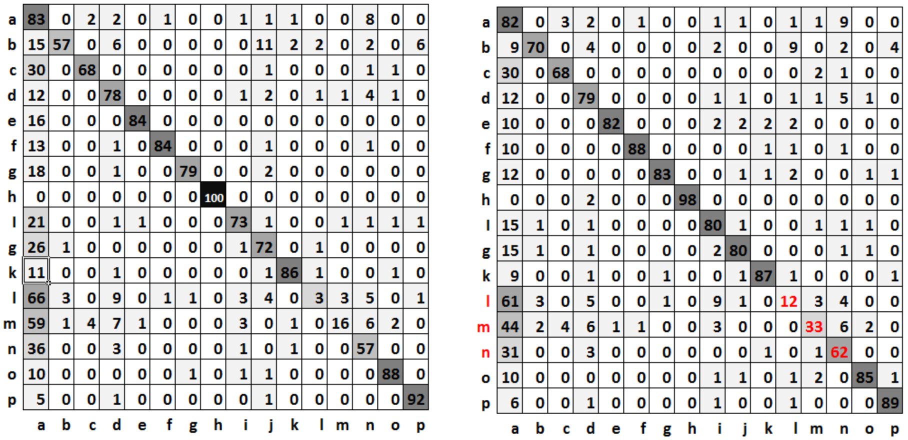
Fig. 6(a) shows the results of Auction(small) dataset. In this case, J48 outperforms SVM and Naïve Bayes. The precision of each classifier is at least 70%. Specifically, the precision of J48 exceeds 80% when the number of features is between 100 to 350. Thus, it can be concluded that J48 can honor the advantage that the size of each brand are uniformly distributed in training data. On the other hand, Fig. 6(b) shows the results of Auction(big) dataset. It can be seen that SVM can achieve the best precision in most cases. Interestingly, the precision of SVM increases from 74.42% to 76.53% while feature number increases from 200 to 500. To find out the root cause, we investigate the features and the classification results when the number of features is 200 and 500. Some critical features appear when the number of features is 500, which do not appear when the number of features is 200. For example, there are 10 more features related to products which is labeled “Maintenance”, such as 風扇(fan), 檢測費(testing fee), 壞掉(out of order), and so on. We can observe the heat maps



(a) Auction (small)

(b) Auction(big)

Fig. 6. Precision with different classifiers: Auction(small) and Auction(big) datasets



	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p
a	83	0	2	2	0	1	0	0	1	1	1	0	0	8	0	0
b	15	57	0	6	0	0	0	0	0	11	2	2	0	2	0	6
c	30	0	68	0	0	0	0	0	0	1	0	0	0	1	1	0
d	12	0	0	78	0	0	0	0	1	2	0	1	1	4	1	0
e	16	0	0	0	84	0	0	0	0	0	0	0	0	0	0	0
f	13	0	0	1	0	84	0	0	0	1	0	0	0	1	0	0
g	18	0	0	1	0	0	79	0	0	2	0	0	0	0	0	0
h	0	0	0	0	0	0	0	100	0	0	0	0	0	0	0	0
l	21	0	0	1	1	0	0	0	73	1	0	0	1	1	1	1
g	26	1	0	0	0	0	0	0	1	72	0	1	0	0	0	0
k	11	0	0	1	0	0	0	0	0	1	86	1	0	0	1	0
l	66	3	0	9	0	1	1	0	3	4	0	3	3	5	0	1
m	59	1	4	7	1	0	0	0	3	0	1	0	16	6	2	0
n	36	0	0	3	0	0	0	0	1	0	1	0	0	57	0	0
o	10	0	0	0	0	0	1	0	1	1	0	0	0	0	88	0
p	5	0	0	1	0	0	0	0	0	1	0	0	0	0	0	92

(a) Number of Features: 200

	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p
a	82	0	3	2	0	1	0	0	1	1	0	1	1	9	0	0
b	9	70	0	4	0	0	0	0	2	0	0	9	0	2	0	4
c	30	0	68	0	0	0	0	0	0	0	0	0	2	1	0	0
d	12	0	0	79	0	0	0	0	1	1	0	1	1	5	1	0
e	10	0	0	0	82	0	0	0	2	2	2	2	0	0	0	0
f	10	0	0	0	0	88	0	0	0	0	1	1	0	1	0	0
g	12	0	0	0	0	0	83	0	0	1	1	2	0	0	1	1
h	0	0	0	2	0	0	0	98	0	0	0	0	0	0	0	0
l	15	1	0	1	0	0	0	0	80	1	0	0	1	1	1	0
g	15	1	0	1	0	0	0	0	2	80	0	0	0	1	1	0
k	9	0	0	1	0	0	1	0	0	1	87	1	0	0	0	1
l	61	3	0	5	0	0	1	0	9	1	0	12	3	4	0	0
m	44	2	4	6	1	1	0	0	3	0	0	0	33	6	2	0
n	31	0	0	3	0	0	0	0	0	0	1	0	1	62	0	0
o	10	0	0	0	0	0	0	0	1	1	0	1	2	0	85	1
p	6	0	0	1	0	0	0	0	1	0	0	1	0	0	0	89

(b) Number of Features: 500

Fig. 7. Heat maps of SVM

in Fig. 7. These features improve the classification result in label "Maintenance Service (m)" and "Other Brands (n)" in SVM. For J48, these features also improve the precision of "Maintenance" class. However, these features diminish the precision of label "Other Brands" class. Consequently, the precision does not increase significantly in J48 when the feature number increases from 200 to 500.

5 Conclusion

In this paper, we target at the problem of identifying the brand name of a product in auction sites. To solve this problem, we first made observations from real datasets in the auction site of Yahoo! Taiwan. The root causes are classified into False-Use Problem, Accessory/Repair-Related Problem, and Brand-Name-Omission Problem. To deal with these three problems, a framework BRF is proposed to retrieve brand names from product of auction sites. BRF first eliminates those brand and model names which may interfere the identification of the actual brand name of a product. Then, BRF represents a product title into a set of representative keywords with their importance by Chi-Square attribute evaluation and TF-IDF. Finally, BRF models the problem of retrieving brand names from the product titles as a classification problem. Extensive experiments are then conducted by using real datasets, and the experimental results showed the effectiveness of BRF.

References

1. Hung, C.-C., Peng, W.-C.: A regression-based approach for mining user movement patterns from random sample data. Data Knowledge Engineering 70(1), 1–20 (2011)
2. Liu, H., Setiono, R.: Chi2: Feature selection and discretization of numeric attributes. In: IEEE 7th International Conference on Tools with Artificial Intelligence, pp. 338–391 (1995)

3. Baye, M.R., De los Santos, B., Wildenbeest, M.R.: The evolution of product search. Journal of Law, Economics & Policy 9(2), 201–221 (2012)
4. Vapnik, V.N.: Statistical Learning Theory. John Wiley & Sons (1998)
5. Han, J., Kamber, M.: Data Mining: Concepts and Techniques, 2nd edn. Morgan Kaufmann (2006)
6. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The WEKA Data Mining Software: An Update. SIGKDD Explorations 11(1) (2009)