I. INTRODUCTION

Recently, a huge number of documents have been produced and stored in digital archives [1]. Therefore, document classification plays a very important role in many information management and retrieval tasks. It refers to the task of classifying a document into a predefined category. Most digital documents are frequently updated, and writers disseminate information and present their ideas on various topics [2]. Unlike news articles written by well-educated journalists and standardized according to official style guides, most digital documents, such as weblogs and twitter tweets, tend to contain colloquial sentences and slang that misleads the classifier, because anybody can publish anything [3-6]. Considering this cumbersomeness of document classification, researchers have proposed the following approaches.

First, naive Bayes-based approaches have performed well from the perspectives of spam filtering and email categorization, and they require a small training corpus to estimate the parameters necessary for document classification [4]. However, the conditional independence assumption is violated by real-world data and does not consider the frequency of word occurrences [7]. Therefore, these approaches perform very poorly when features are highly correlated [8].

Second, the support vector machine (SVM)-based approaches have been recognized as some of the most effective document classification methods as compared to supervised machine learning algorithms. Nevertheless, these approaches have certain difficulties with respect to parameter tuning and kernel selection [7]. Further, the performance of SVM suitable for binary classification...
degrades rapidly, as the amount of training data decreases, resulting in a relatively poor performance on large-scale datasets with many labels [8].

Third, knowledge-based approaches have utilized knowledge, such as the meaning and relationships of the words, which can be obtained from an ontology, such as WordNet [8]. However, considering the restricted incompleteness of an ontology, the effect of word sense and relation disambiguation is quite limited [9, 10].

Fourth, Web-based approaches have been used for mining desired information, such as popular opinions from Web documents obtained by traversing the Web in order to explore information that is related to a particular topic of interest. However, Web crawlers attempt to search the entire Web, which is impossible because of the size and the complexity of the World Wide Web [11].

In this paper, we propose a document classification model to solve the imbalance of the training corpus size per category by utilizing Web documents. The rest of this paper is organized as follows: Section II presents an overview of the proposed model consisting of a prediction phase to assign a category to a given document, and a learning step to balance the training corpus size by adding Web documents. Then, Section III presents some experimental results, and the characteristics of the proposed method are presented as the conclusion of this paper in Section IV.

II. WEB-BASED DOCUMENT CLASSIFICATION MODEL

As shown in Fig. 1, the proposed Web-based document classification model is composed of a prediction phase and a learning phase. Given an unlabeled document in the prediction phase, the feature extraction step extracts some word features, such as unigrams and bigrams from the document. Then, the document classification step predicts the category of the document by utilizing the learned parameters. On the other hand, the learning phase obtains these learned parameters, such as word-frequency distribution from the category-labeled documents. Section II-A introduces the document classification method, and Section II-B describes the proposed Web document acquisition method in detail.

A. Document Classification Method

The proposed document classification method selects the category $c_i$, taking the highest matching probability for the given unlabeled document $D$, as represented in Eq. (1), where $c_i$ denotes an element of the category set $C$. In order to make it easy to mathematically deal with the document classification problem, the proposed model represents the unlabeled document $D$ as the document vector $d$ consisting of feature values extracted from the document by the feature extraction step. Each feature value indicates the number of times that each word feature, such as the bigram “arcade game” appears in the document [12]. Most elements of the document vector take a zero value.

$$\arg\max_{c_i \in C} P(c_i \mid D) = \arg\max_{c_i \in C} P(c_i \mid \tilde{d}).$$

In the maximum entropy framework [12-15], the conditional probability of predicting an outcome $c_i$ given the history $d$ is defined as follows:

$$P(c_i \mid \tilde{d}) = \frac{1}{Z(d)} \exp \left( \sum_{j=1}^{k} \lambda_j f_j(\tilde{d}, c_i) \right),$$

$$\sum_{c_i} P(c_i \mid \tilde{d}) = 1,$$
Table 1. Some word features selected on the basis of chi-square statistics

<table>
<thead>
<tr>
<th>Health</th>
<th>Shopping</th>
<th>SNS</th>
<th>Sports</th>
</tr>
</thead>
<tbody>
<tr>
<td>health</td>
<td>shopping</td>
<td>chatting</td>
<td>baseball</td>
</tr>
<tr>
<td>diet</td>
<td>barcode</td>
<td>Twitter</td>
<td>football</td>
</tr>
<tr>
<td>stretching</td>
<td>coupons</td>
<td>ICQ</td>
<td>league</td>
</tr>
<tr>
<td>calories</td>
<td>price</td>
<td>guest book</td>
<td>billiards</td>
</tr>
<tr>
<td>diet dance</td>
<td>QR code</td>
<td>Myspace</td>
<td>soccer</td>
</tr>
</tbody>
</table>

SNS: social network service.

where \( f(d,c) \) denotes the feature function, \( \lambda_i \) represents the weighting parameter of \( f(d,c) \), \( k \) refers to the number of features, and \( Z(d) \) indicates the normalization factor for Eq. (3) \([13-15]\).

B. Web Document Acquisition Method

In order to balance the training corpus size per category, the proposed Web-based document classification model increases the number of labeled documents by utilizing a Web search engine to retrieve the Web documents.

In the query generation step shown in Fig. 1, the proposed method first selects some useful word features per category on the basis of the chi-square statistics \([16, 17]\). The chi-square statistics of word feature \( f_j \) in the category \( c_i \) is defined as follows:

\[
\chi^2(f_j, c_i) \approx \frac{N(AD - CB)^2}{(A + C)(B + D)(A + B)(C + D)},
\]

where \( A \) denotes the number of documents containing the word feature \( f_j \) in the category \( c_i \), \( B \) represents the number of documents containing the word feature \( f_j \) in other categories rather than \( c_i \), \( C \) indicates the number of documents not containing the word feature \( f_j \) in the category \( c_i \), \( D \) refers to the number of documents not containing the word feature \( f_j \) in other categories rather than \( c_i \), and \( N \) denotes the total number of documents. Each word feature \( f_j \) is computed for every category, and the word features with top \( n \) highest chi-square statistics are used for the query candidates. Table 1 shows some examples of word features selected for the categories, such as health, music, shopping, social network service (SNS), and sports according to the chi-square statistics.

Given the selected word features per category, the proposed Web document acquisition method generates a query for the Web search engine. Considering that the lower-ranked word features can be less related to the characteristics of the category, the proposed method combines two of the higher-ranked word features and the title of the category for the purpose of retrieving the Web documents closely related to the category. Therefore, the query consists of three word features: the category title, one of the top 5 ranked word features, and one of the top 200 ranked word features except the top 5 ranked word features. For each category, the proposed query generation step maximally yields 975 queries by multiplying the 5 higher-ranked word features with the 195 lower-ranked word features.

Finally, the proposed Web document acquisition method sends each combined query to the open application programming interface (API) of the Web search engine, and receives the snippet results retrieved from the Web search engine. The proposed Web document acquisition method assumes the snippet results of each query as one Web document.

III. EXPERIMENTS

In order to prove the validity of utilizing the Web documents, we have tested the MALLET document classification package \([12]\) with a mobile application description document corpus \([18]\), which is divided into 90% for the training set and 10% for the test set. The document corpus consists of 3,521 documents and 302,772 words. Each
document has 3 to 515 words, and a document is composed of 85.99 words on average. As described in Fig. 2, the distribution of documents is not balanced among categories in the corpus. For instance, the documents corresponding to the Utility category capture roughly 21% of the corpus, while the documents corresponding to the music category capture 0.28% of the corpus.

On the other hand, the proposed method is evaluated on the basis of the following evaluation criteria: precision, recall, and F-measure. Precision indicates the ratio of correct candidate categories to the candidate categories predicted by the proposed document classification model. Recall indicates the ratio of correct candidate categories to the total number of categories of the documents in the corpus. F-measure indicates the harmonic mean of the precision and the recall.

Fig. 3 shows that the performance is improved by adding Web documents; here, each performance value is an average of the 10-fold cross validation. In this figure, the y-axis indicates the recall value, and x-axis denotes the number of Web documents. Because the proposed document classification model predicts all categories of the given documents, the precision is the same as the recall. The baseline performance without any document classification model is 21% because the documents corresponding to the utility account for roughly 21% of the corpus.

Equal size performance indicates the recall of the document classification model MALLET, learned from the training set with a different number of Web documents per category for the balance of the training corpus size. Equal addition performance indicates the recall of the document classification model MALLET, learned from the training set with an equal number of Web documents per category [19]. For example, the Equal size method does not add any Web documents for categories, such as Utility that already have many documents, while the method adds many Web documents for categories, such as Music that have few documents. On the other hand, the Equal addition method always adds the same number of Web documents to each category according to the x-axis value.

Fig. 3 shows that the Equal size method is slightly more effective than the Equal addition method [19] because the document classification model decreases the tendency of the ambiguous or less informative document to correspond to the category with the relatively more documents, by balancing the number of documents per category. Further, Fig. 3 shows that the performance does not always increase according to the addition of Web documents because the characteristics of mobile application description documents is considerably different from the characteristics of Web documents, which are very sensitive to Web search queries. Moreover, the performance does not generally increase much upon the addition of Web documents as most category prediction results in the test set are biased towards a few categories corresponding to many documents, such as Utility category, Puzzle/Board game category, and Health/Fitness category, whereas Web documents improve the classification performance for the categories with a few documents, such as Music category, Testyourself category, and SNS category.
Fig. 4 shows the effects of Web documents according to categories at the peak performance by adding 150 Web documents per category. It indicates that the effects of Web documents can differ considerably from each other according to the category, although the general performance did not increase much upon the addition of Web documents. In particular, the recall and the precision of Music category increased by 100% because the document classification model without Web documents did not predict Music category at all, whereas the document classification model with Web documents attempted to predict the correct Music category once. Like the Testurselv category and the SNS category, categories with fewer mobile application description documents increased the recall because the probability of predicting these categories increased by adding Web documents corresponding to the categories. On the other hand, categories, such as Utility category and the Health/Fitness category with many documents, increased the precision because the possibility of predicting these categories decreases. However, Food category and Education category decreased the recall because the proposed method did not find an appropriate query for Web document retrieval because these categories had few documents.

IV. CONCLUSIONS

In this paper, we propose a document classification model utilizing Web documents. The proposed model has the following characteristics: the proposed model can adjust the balance of the training corpus size per category by adding Web documents to the training corpus. Further, the proposed model can retrieve Web documents that are closely related to each category by combining two of the higher-ranked word features and the category title according to the matching score between the word features and the category. Experimental results show that the proposed model improves performance in some categories in the case of small training sets by balancing the number of documents per category. In the future, we intend to compare the effects of some feature selection methods, such as mutual information and information gain. Further, we plan to automatically generate word clusters and apply them to the proposed model.

ACKNOWLEDGMENTS

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2012R1A1A3013405). Further, this paper is an expanded version of the paper “Application of Web Search Results for Document Classification” published in the Proceedings of ICFICE 2013.

REFERENCES


So-Young Park
received her B.S. in Computer Science and Engineering from Sangmyung University, Korea, in 1997, and her M.S. and Ph.D. in Computer Science from Korea University, Seoul, Korea, in 1999 and 2005, respectively. She has been working as an associate professor at Sangmyung University, Seoul, Korea, since 2007. Her research interests include natural language understanding and data mining.

Juno Chang
received his B.S., M.S., and Ph.D. in Computer Science from Seoul National University, Seoul, Korea, in 1990, 1992, and 1998, respectively. He was a consulting director in i2 Technologies from 1998 to 2003. Since 2003, he has been working as a professor at Sangmyung University, Seoul, Korea. His areas of research include business software, middleware, and digital storytelling.

Taesuk Kihl
received her B.S., M.S., and Ph.D. in Korean Language and Literature from Yonsei University, Seoul, Korea in 1990, 1992, and 2002, respectively. Since 2011, she has been working as an assistant professor at Sangmyung University, Seoul, Korea. Her major field of study is digital storytelling.