A Term Association Translation Model for Naive Bayes Text Classification

Meng-Sung Wu and Hsin-Min Wang

Institute of Information Science, Academia Sinica, Taipei, Taiwan
{wums,whm}@iis.sinica.edu.tw

Abstract. Text classification (TC) has long been an important research topic in information retrieval (IR) related areas. In the literature, the bag-of-words (BoW) model has been widely used to represent a document in text classification and many other applications. However, BoW, which ignores the relationships between terms, offers a rather poor document representation. Some previous research has shown that incorporating language models into the naive Bayes classifier (NBC) can improve the performance of text classification. Although the widely used N-gram language models (LM) can exploit the relationships between words to some extent, they cannot model the long-distance dependencies of words. In this paper, we study the term association modeling approach within the translation LM framework for TC. The new model is called the term association translation model (TATM). The innovation is to incorporate term associations into the document model. We employ the term translation model to model such associative terms in the documents. The term association translation model can be learned based on either the joint probability (JP) of the associative terms through the Bayes rule or the mutual information (MI) of the associative terms. The results of TC experiments evaluated on the Reuters-21578 and 20newsgroups corpora demonstrate that the new model implemented in both ways outperforms the standard NBC method and the NBC with a unigram LM.

Keywords: Term association, mutual information, Bayes, translation language model, text classification.

1 Introduction

Text classification (TC) is the task of classifying documents into a set of predefined categories. It has long been an important research topic in information retrieval (IR). Many statistical classification methods and machine learning (ML) techniques have been developed to TC, such as the naive Bayes classifier [12], the support vector machines [10], the k-nearest neighbor method [20], and the boosting method [16]. In addition, text classification based on term associations [1] is also a promising approach. The performance of text classification highly depends on the document representation. Most of the existing methods represent a document using a vector space model (VSM) or a language model (LM).

Generally, the bag-of-words (BoW) method is a widely used data representation in IR and TC. Under this scheme, each document is modeled as a vector with a dimension equal to the size of the dictionary, and each element of the vector denotes the frequency that a word appears in the document. Basically, all the words are treated independently.

One of the important restrictions in most of the existing TC methods may lie in that the individual terms are usually too general and that these methods do not consider the associations between words in the documents. In some cases of TC, individual words are not sufficient to represent the accurate information of the document. For example, a document with "shuttle launch" may be assumed to belong to the "ball game" class. However, if the word "NASA" is an association term, it is very likely that the document should be assigned to the "aeronautics" class.

It is well-known that the relationships between words are very important for statistical language modeling. Using LM for TC has been studied recently [2,14]. Although N-gram LM can exploit the relationships between words, they only consider the dependencies of neighboring words [5]. For example, the trigram LM is unable to characterize word dependence beyond the span of three successive words. In [22], the trigram LM was improved by integrating with the trigger pairs, which extract the word relationships from the sequence of historical words. Nevertheless, a trigger pair is word order dependent. In other words, a word can only be triggered by the previous context. Recent studies have revealed that modeling term associations could provide richer semantics of documents for LM and IR [4,18,19]. Cao et al. [4] integrated the word co-occurrence information and the WordNet information into language models. Wei and Croft [18] investigated the use of term associations to improve the performance of LM-based IR. In [19], the word associations were integrated into the topic modeling paradigm. Adding word associations to represent a document inevitably increases the model's complexity, but the new information reduces the ambiguity mentioned above. Generally, any set of words co-occur in the contexts can be considered having a strong association and collected as the associative words, e.g., "uneven bars" and "balance" in the class of gymnastics and "aerofoil" and "jet engine" in the class of airplane transportation. However, the associative words are not necessary to co-occur in a document. We believe that a language model considering term associations would be definitely more useful in TC.

In this paper, we propose a novel model for text classification by incorporating the strengths of term associations into the translation LM framework. Different from the traditional TC techniques and algorithms in the literature, we model the associations between words existing in the documents of a class. To discover the associative terms in the documents, we learn the translation language model based on the joint probability (JP) of the associative terms through the Bayes rule and based on the mutual information (MI) of the associative terms.

The remainder of this paper is organized as follows. In Section 2, we briefly review the framework of the naive Bayes classifier and language models. The proposed models for text classification are presented in Section 3. Experimental setup and results are discussed in Section 4. Finally, we give the conclusions in Section 5.

2 Related Work

2.1 Terminology

We begin by defining the notation and terminology in this paper. A **word** or **term** is a linguistic building block for text. A word is denoted by $w \in V = \{1, 2, \ldots, |V|\}$, where $|V|$ is number of distinct words/terms. A **document**, represented by $\mathbf{d} = \{w_1, \cdots, w_{n_d}\}$, is an ordered list of n_d words. A query, denoted by $\mathbf{q} = \{q_1, \cdots, q_T\}$, is a string of T words. A **collection** of documents is denoted by $D = \{\mathbf{d}_1, \cdots, \mathbf{d}_{|D|}\}$, where $|D|$ is the number of documents in collection D. A **background model**, denoted by $\mathcal{M}_B$, is the language model estimated in collection D. A set of **class labels** is denoted by $C = \{c_1, \cdots, c_{|C|}\}$, where $|C|$ is the number of distinct classes. A **LM** $\mathcal{M}$ is a probability function defined on a set of word strings. This includes the important special case of the probability $P(w|\mathcal{M})$ of a word w. A **class LM**, denoted by $\mathcal{M}_C$, is the language model estimated based on class c.

2.2 Naive Bayes Classifier

The naive Bayes classifier (NBC) is a popular machine learning technique for text classification. The method assumes a probabilistic generative model for text. A common and simple representation of a document in TC is the bag of words (BoW) model. The model ignores the word order and just captures the number of occurrences of each word in the document. The NBC classifies a document through two stages: the learning stage and the classifying stage. It is assumed that the probability of each word in a document is independent to that of other words, and each document is drawn from a multinomial distribution of words. In the learning stage, the naive Bayes classifier estimates the conditional probability $P(c|d)$, which represents the probability that a document d belongs to a class c. Using the Bayes rule, we have

$$P(c|d) = \frac{P(d|c)P(c)}{P(d)} = \frac{P(d|c)P(c)}{\sum_c P(d|c)P(c)}, \tag{1}$$

where $P(d|c)$ is the likelihood of document d under class c. By assuming that all words in d are independent of each other, $P(d|c)$ can be further decomposed into the product of individual feature (word) probabilities as follows

$$P(d|c) = \prod_{w \in d} P(w|c). \tag{2}$$

The word probability $P(w|c)$ and the class prior probability $P(c)$ are estimated from the training documents with Laplace smoothing as follows

$$P(w|c) = \frac{1 + n(w,c)}{|V| + N(c)}, \tag{3}$$

$$P(c) = \frac{1 + n(d,c)}{|C| + |D|}, \tag{4}$$

where $n(w,c)$ is the number of times word w occurs in the training documents that belong to class c; $N(c)$ is the total number of words in the training documents that belong to class c; $n(d,c)$ denotes the number of documents that belong to class c; and $|D|$ is total number of training documents.

Several extensions of the naive Bayes classifier have been proposed. For example, Nigam et al. [13] combined the Expectation-Maximization (EM) algorithm and the naive Bayes classifier to learn from both labeled and unlabeled documents in a semi-supervised manner. More recently, Dai et al. [7] proposed a transfer learning algorithm to learn the naive Bayes classifier for text classification, which allowed the distributions of the training and test data to be different. However, these methods all assume that the words in a document are independent of each other; hence, they cannot cope well with the term dependence and association.

2.3 Language Models for Information Retrieval

Statistical language modeling plays an important role in automatic speech recognition (ASR) and IR. Most ASR systems are built by combining the N-gram language model and the acoustic hidden Markov model (HMM) to predict the best word sequence corresponding to an input speech utterance. In an IR system, the word sequence of an input query is adopted to retrieve the relevant text documents. In Ponte and Croft's work that applied LM in IR [15], the retrieval performance was improved by statistical modeling of natural language. According to the maximum a posteriori decision rule, the ranking function $f(\cdot)$ is established as a posterior probability,

$$\hat{\mathbf{d}} = \arg\max_{\mathbf{d}_m} f(\mathbf{q}, \mathbf{d}_m) = \arg\max_{\mathbf{d}_m} P(\mathbf{d}_m|\mathbf{q}) = \arg\max_{\mathbf{d}_m} P(\mathbf{q}|\mathbf{d}_m)P(\mathbf{d}_m). \tag{5}$$

Assuming that the documents $\{\mathbf{d}_1, \cdots, \mathbf{d}_{|D|}\}$ have an equal prior probability of relevance, the ranking can be done according to the likelihood of the N-gram language model

$$P(\mathbf{q}|\mathbf{d}_m) = P(q_1, \cdots, q_T|\mathbf{d}_m) = \prod_{t=1}^{T} P(q_t|q_{t-n+1}^{t-1}, \mathbf{d}_m), \tag{6}$$

where each word q_t only depends on its $n-1$ historical words $q_{t-n+1}^{t-1} = \{q_{t-n+1}, \cdots, q_{t-1}\}$. $P(q_t|q_{t-n+1}^{t-1}, \mathbf{d}_m)$ can be estimated according to the maximum likelihood (ML) criterion as follows,

$$P_{\mathrm{ML}}(q_t|q_{t-n+1}^{t-1},\mathbf{d}_m)=\frac{c(q_{t-n+1}^{t},\mathbf{d}_m)}{c(q_{t-n+1}^{t-1},\mathbf{d}_m)}, \tag{7}$$

where $c(q_{t-n+1}^{t},\mathbf{d}_m)$ denotes the number of times that word q_t follows the historical words q_{t-n+1}^{t-1} in document $\mathbf{d}_m$ and $c(q_{t-n+1}^{t-1},\mathbf{d}_m)$ denotes the number of times that the historical words q_{t-n+1}^{t-1} occur in document $\mathbf{d}_m$. The unigram document model is generally adopted in the IR community [15]. However, the document terms are often too few to train a reliable ML-based model because the unseen words lead to zero unigram probabilities. Zhai and Lafferty [21] have used several smoothing methods to deal with the data sparseness problem in LM-based IR.

Since previous research [4,18,19] has shown that some relationships exist between words, we utilize them in the document model rather than using the traditional unigram document model for text classification.

3 The Term Association Translation Models

3.1 Language Models for Text Classification

LM was first introduced to TC by Peng and Schuurmans [14]. The score of a class c for a given document d can be estimated by (1). Then, the class of the document can be decided as follows

$$c^*=\arg\max_{c\in C}P(c|d)=\arg\max_{c\in C}P(d|c)P(c). \tag{8}$$

Assuming that $P(c)$ is uniformly distributed and applying the unigram class LM in the task, the decision can be rewritten as

$$\begin{aligned}c^*&=\arg\max_{c\in C}P(d|c)\\&=\arg\max_{c\in C}\prod_{i=1}^{n_d}P(w_i|c).\end{aligned} \tag{9}$$

The traditional naive Bayes classifier usually uses Laplace smoothing to deal with the zero probability problem. However, some previous research has shown that it is not as effective as the smoothing methods for language modeling [2,14]. Therefore, we can interpolate a unigram class LM with the unigram collection background model by using the *Jelinek-Mercer* smoothing method as follows,

$$P(w_i|\mathcal{M}_C)=\lambda P(w_i|c)+(1-\lambda)P(w_i|\mathcal{M}_B), \tag{10}$$

where λ can be tuned empirically. In this paper, the method based on (10) is denoted as NBC-UN, and λ is set to 0.5.

In order to discover the association between two terms w_i and w, we are interested in $P_t(w_i|w)$, the probability that word w_i will occur given that w

occurs. The term translation probability $P_t(w_i|w)$ is different from the bigram probability $P(w_i|w)$ in that the words w_i and w are not limited to occur in order and adjacently in the former. Then, the term association information can be integrated into the unigram class model as follows,

$$P(w_i|c) = \sum_{w \in c} P_t(w_i|w)P(w|c), \tag{11}$$

where $P(w|c)$ reflects the distribution of words in the training documents of class c, which can be computed via the maximum likelihood estimate. By replacing $P(w_i|c)$ in (10) with the one computed by (11), we have

$$P(w_i|\mathcal{M}_C) = \lambda[\sum_{w \in c} P_t(w_i|w)P(w|c)] + (1-\lambda)P(w_i|\mathcal{M}_B). \tag{12}$$

The model in (12) is obviously more computationally intensive than the model in (10). Therefore, we need to build a global term translation model for all classes and the word probability distribution for each class beforehand. To discover the associative terms in the training documents, we learn the translation LM based on the joint probability of the associative terms through the Bayes rule and based on the mutual information (MI) of the associative terms.

3.2 Translation Model Estimation Using Joint Probability Model

This section describes our first way of constructing the term translation probability $P_t(w_i|w)$. By definition, we can express the conditional probability as the joint probability of words w_i and w over the probability of word w

$$P_t(w_i|w) = \frac{P(w_i, w)}{P(w)}, \tag{13}$$

where the join probability of w_i and w can be expressed as

$$P(w_i, w) = \sum_{c} P(w_i, w|c)P(c) = \sum_{c} P(w_i|c)P(w|c)P(c), \tag{14}$$

if w_i and w are assumed sampled independently and identically from the unigram class model c, and the probability of w can be expressed as

$$P(w) = \sum_{c} P(w|c)P(c). \tag{15}$$

After re-normalizing $P(w_i, w)$ in (14) and $P(w)$ in (15), and considering a uniform prior $P(c)$, we obtain

$$P_t(w_i|w) = \frac{\sum_c P(w_i|c)P(w|c)}{\sum_c P(w|c)}. \tag{16}$$

The method based on (12) with $P_t(w_i|w)$ computed by (16) is denoted as TATM-JP (the *term association translation model* estimated by the *joint probability* of terms).

3.3 Translation Model Estimation based on Mutual Information

Our second way of constructing the term translation probability $P_t(w_i|w)$ is based on the mutual information (MI). In information theory, the MI of two random variables is a quantity that measures their mutual dependence. MI is a good measure to assess how two words are related to each other [6,22]. We use the average mutual information (AMI) [22] to measure the strength of the association between words w_i and w. The AMI between w_i and w is defined as follows

$$\begin{aligned} AMI(w_i, w) = P(w_i, w) log\frac{P(w_i, w)}{P(w_i)P(w)} + P(w_i, \bar{w}) log\frac{P(w_i, \bar{w})}{P(w_i)P(\bar{w})} \\ + P(\bar{w}_i, w) log\frac{P(\bar{w}_i, w)}{P(\bar{w}_i)P(w)} + P(\bar{w}_i, \bar{w}) log\frac{P(\bar{w}_i, \bar{w})}{P(\bar{w}_i)P(\bar{w})} \end{aligned} \quad (17)$$

where $P(w_i, w)$ is estimated as the ratio of the number of documents that contain both w_i and w, i.e., $c_d(w_i, w)$, and the total number of documents $|D|$ as follows

$$P(w_i, w) = \frac{c_d(w_i, w)}{|D|}; \quad (18)$$

$P(w_i, \bar{w})$ is computed by

$$P(w_i, \bar{w}) = \frac{c_d(w_i) - c_d(w_i, w)}{|D|}, \quad (19)$$

where $c_d(w_i)$ is the number of documents that contain w_i; $P(w)$ is estimated as the ratio of the number of documents that contain w and the total number of documents; $P(\bar{w})$ is estimated as the ratio of the number of documents that do not contain w and the total number of documents; and the other probabilities are estimated in a similar way. According to [11], the term translation probability $P_t(w_i|w)$ can be calculated by normalizing the mutual information score as follows

$$P_t(w_i|w) = \frac{AMI(w_i, w)}{\sum_{w_j} AMI(w_j, w)}. \quad (20)$$

If the two words w_i and w tend to associate with each other, the probability would be higher. The method based on (12) with $P_t(w_i|w)$ computed by (20) is denoted as TATM-MI (the *term association translation model* estimated based on the *mutual information* of terms).

4 Experiments

4.1 Corpora

We evaluate the proposed TC methods on two standard document collections: Reuters-21578 (Reuters)[1] and 20 Newsgroups (20NG)[2]. According to the ModApte

[1] http://kdd.ics.uci.edu/databases/reuters21578/reuters21578.html

[2] http://people.csail.mit.edu/jrennie/20Newsgroups/

Table 1. Statistics of the Reuters collection

Category	Training Set	Test Set
earn	2877	1087
acq	1650	719
money-fx	538	179
grain	433	149
crude	389	189
trade	369	118
interest	347	131
wheat	212	71
ship	197	89
corn	182	56

split, the Reuters corpus is separated into 7,194 documents for training and 2,788 documents for testing. 135 categories have been defined, but only 118 categories have documents assigned to them. Following Debole and Sebastiani's work in [8], we consider the most frequent ten categories in the experiments. The 10 categories and the numbers of documents used for training and testing in each category are listed in Table 1. The 20NG dataset is a collection of 19,974 documents collected from 20 different newsgroups. We consider the 20 newsgroups as the 20 categories. For each category, we randomly select 60% of the documents for training and the remaining 40% for testing. Since the 20NG collection distributes roughly evenly across 20 newsgroups, each category has almost the same number of training (or testing) documents.

4.2 Performance measure

In the following experiments, the performance of text classification is evaluated in terms of the recall (R), precision (P), and F-measure (F), calculated as follows:

$$\text{recall} = \frac{\#\text{ of correct postive predictions}}{\#\text{ of postive examples}}, \tag{21}$$

$$\text{precision} = \frac{\#\text{ of correct postive predictions}}{\#\text{ of postive predictions}}, \tag{22}$$

$$F = \frac{2 \times \text{recall} \times \text{precision}}{\text{recall} + \text{precision}}. \tag{23}$$

To evaluate the average performance across classes, we use the micro-averaged score and macro-averaged score [20]. The micro-averaged score is calculated by mixing together the documents across all the classes. The macro-averaged score is obtained by taking the average of the recall, precision, and F-measure values for each category

Table 2. Experimental results (in F-measure) for the Reuters collection

	NBC	NBC-UN	TATM-JP	TATM-MI
earn	0.814	0.825	0.819	0.824
acq	0.801	0.811	0.801	0.802
money-fx	0.511	0.521	**0.543**	0.539
grain	0.578	0.583	0.616	**0.634**
crude	0.577	0.596	0.610	**0.618**
trade	0.439	0.434	**0.477**	0.465
interest	0.483	0.477	**0.516**	0.502
wheat	0.490	0.506	0.577	**0.603**
ship	0.571	0.583	0.624	**0.639**
corn	0.466	0.468	**0.527**	0.491
micro-averaged	0.709	0.720	0.727	**0.731**

Table 3. The micro/macro-averaged precision, recall, and F-measure of different methods evaluated on the 20NG dataset

	Micro-averaged			Macro-averaged		
	P	R	F	P	R	F
NBC	0.802	0.800	0.801	0.817	0.795	0.806
NBC-UN	0.809	0.807	0.808	0.822	0.802	0.812
TATM-JP	0.818	0.815	0.817	0.827	0.810	0.818
TATM-MI	0.821	0.819	0.820	0.829	0.814	0.821

4.3 Experimental results

We compare our term association translation models (TATM-JP and TATM-MI) with the naive Bayes classifier with Laplace smoothing (NBC) and the naive Bayes classifier with the unigram language model (NBC-UN).

Table 2 shows the results of text classification experiments evaluated on the Reuters collection. The measure used is the F-measure on the ten most populated Reuters-21578 categories and the micro-averaged F-measure (micro-F) over all categories. Comparing the results of NBC and NBC-UN, it is obvious that using language models improves the classification effectiveness of the naive Bayes classifier. Both proposed methods consistently outperform NBC and respectively perform better than NBC-UN in four out of ten categories. The micro-F of TATM-MI is 0.731, which is better than that of TATM-JP (0.727), NBC-UN (0.720) and NBC (0.709). The relative improvement in micro-F by TATM-MI is 3.1% over NBC and 1.5% over NBC-UN.

Table 3 shows the experimental results for the 20NG dataset in terms of the micro/macro-averaged precision, recall, and F-measure. The micro-F is 0.817 for TATM-JP and 0.82 for TATM-MI, which are better than that obtained by NBC (0.801) and NBC-UN (0.808). The relative improvements by TATM-JP over NBC and NBC-UN are 2% and 1.11%, respectively. Similarly, the relative improvements by TATM-MI over NBC and NBC-UN are 2.37%, and 1.49%, respectively. The improvements of TATM-JP and TATM-MI over NBC and NBC-UN are statistically significant according to the t-test. In addition, learning the term association translation model based on the mutual information for all data sets is more efficient than learning the term association translation model by the joint probability. As expected, the performance in micro-F on the 20NG dataset is very similar to that in macro-F because each class has a similar number of training and testing documents. Again, we can see that TATM-MI performs the best.

Several observations can be drawn from the results. First, the performance of text classification can be improved by incorporating language models into the naive Bayes classifier. Second, the proposed document model with term association modeling leads to improvements over NBC and NBC-UN. The new model could be applied to other topic document models.

5 Conclusion and Future Work

The use of term associations for TC has attracted great interest. This paper has presented a new term association translation model, which models term associations, for TC. The proposed model can be learned based on the joint probability of the associative terms through the Bayes rule or based on the mutual information of the associative terms. The experimental results show that the new model learned in either way outperforms the traditional TC methods. For future work, we plan to investigate the effect of the feature selection method [17] for the selection of associative terms. In addition, we will integrate our model into the topic models such as probability latent semantic analysis (PLSA) [9] or latent Dirichlet allocation (LDA) [3] for text classification. Another interesting direction is to combine the term association document model with the relevance-based document model, and apply the combined model in TC.

References

1. Antonie, M.L., Zaiane, O.R.: Text Document Categorization by Term Association. In: Proceedings of IEEE 2002 International Conference on Data Mining (ICDM). pp. 19–26 (2002)
2. Bai, J., Nie, J.Y.: Using language models for text classification. In: Proceedings of the Asia Information Retrieval Symposium (AIRS) (2004)
3. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent Dirichlet Allocation. Journal of Machine Learning Research 3(4-5), 993–1022 (2003)
4. Cao, G., Nie, J.Y., Bai, J.: Integrating word relationships into language models. In: Proceedings of the ACM SIGIR International Conference on Research and Development in Information Retrieval. pp. 298–305 (2005)
5. Chien, J.T., Wu, M.S., Peng, H.J.: Latent semantic language modeling and smoothing. International Journal of Computational Linguistics and Chinese Language Processing 9(2), 29–44 (2004)
6. Church, K.W., Hanks, P.: Word association norms, mutual information, and lexicography. Computational Linguistics 16(1), 22–29 (1990)

7. Dai, W., Xue, G.R., Yang, Q., Yu, Y.: Transferring Naive Bayes Classifiers for Text Classification. In: Proceedings of AAAI conference on Artificial intelligence. pp. 540–545 (2007)
8. Debole, F., Sebastiani, F.: An analysis of the relative difficulty of Reuters-21578 subsets. Journal of the American Society for Information Science and Technology 56(2), 584–596 (2005)
9. Hofmann, T.: Probabilistic latent semantic indexing. In: Proceedings of ACM SIGIR International Conference on Research and Development in Information Retrieval. pp. 50–57 (1999)
10. Joachims, T.: Text Categorization With Support Vector Machines: Learning With Many Relevant Features. In: Proceedings of European Conference on Machine Learning (ECML). pp. 137–142 (1998)
11. Karimzadehgan, M., Zhai, C.: Estimation of Statistical Translation Models Based on Mutual Information for Ad Hoc Information Retrieval . In: Proceedings of ACM SIGIR International Conference on Research and Development in Information Retrieval. pp. 323–330 (2010)
12. McCallum, A., Nigam, K.: A Comparison of Event Models for Naive Bayes Text Classification. In: AAAI-98 Workshop on Learning for Text Categorization. pp. 41–48 (1998)
13. Nigam, K., Mccallum, A.K., Thrun, S., Mitchell, T.: Text Classification from Labeled and Unlabeled Documents using EM . Machine Learning 39(2/3), 103–134 (2000)
14. Peng, F., Schuurmans, D.: Combining Naive Bayes and n-Gram Language Models for Text Classification. In: Proceedings of the 25th European Conference on Information Retrieval Research (ECIR). pp. 335–350 (1994)
15. Ponte, J.M., Croft, W.B.: A language modeling approach to information retrieval. In: Proceedings of ACM SIGIR International Conference on Research and Development in Information Retrieval. pp. 275–281 (1998)
16. Schapire, R.E., Singer, Y.: BoosTexter: A Boosting-based System for Text Categorization. Machine Learning 39(2/3), 135–168 (2000)
17. Schneider, K.M.: Weighted Average Pointwise Mutual Information for Feature Selection in Text Categorization. In: Proceedings of the European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD). pp. 252–263 (2005)
18. Wei, X., Croft, W.B.: Modeling term associations for ad-hoc retrieval performance within language modeling framework. In: Proceedings of the 29th European conference on IR research. pp. 52–63 (2007)
19. Wu, M.S., Lee, H.S., Wang, H.M.: Exploiting semantic associative information in topic modeling. In: Proceedings of the IEEE Workshop on Spoken Language Technology. pp. 384–388 (2010)
20. Yang, Y.: An Evaluation of Statistical Approaches to Text Categorization . Journal of Information Retrieval 1(1-2), 67–88 (1999)
21. Zhai, C., Lafferty, J.: A study of smoothing methods for language models applied to Ad Hoc information retrieval. In: Proceedings of ACM SIGIR International Conference on Research and Development in Information Retrieval. pp. 334–342 (2001)
22. Zhou, G., Lua, K.: Interpolation of n-gram and mutual-information based trigger pair language models for Mandarin speech recognition. Computer Speech and Language 13(2), 125–141 (1999)