Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2015, Article ID 985204, 5 pages
http://dx.doi.org/10.1155/2015/985204

Research Article

Error Checking for Chinese Query by Mining Web Log

Jianyong Duan,[1] Peng Mi,[1] and Hui Liu[2]

[1]*College of Computer Science, North China University of Technology, No. 5 Jinyuanzhuang Road, Shijingshan District, Beijing 100144, China*
[2]*School of Business Information, Shanghai University of International Business and Economics, No. 1900 Wenxiang Road, Shanghai 201620, China*

Correspondence should be addressed to Jianyong Duan; duanjy@hotmail.com

Received 13 March 2015; Accepted 16 July 2015

Academic Editor: Chih-Cheng Hung

For the search engine, error-input query is a common phenomenon. This paper uses web log as the training set for the query error checking. Through the n-gram language model that is trained by web log, the queries are analyzed and checked. Some features including query words and their number are introduced into the model. At the same time data smoothing algorithm is used to solve data sparseness problem. It will improve the overall accuracy of the n-gram model. The experimental results show that it is effective.

1. Introduction

Recently, with the development of search engine technology, abundant web log is produced. These resources are naturally used into query error checking and correcting. The error checking is the crucial step for query correction. Chinese and English belong to different language families; their queries have differences among query length, spelling error, and so forth. The average lengths of queries are 1.85 words in Chinese as well as 2.35 words in English [1]. In English query log, there are nearly 10% ~15% of queries with the spelling error. Although there are no spelling errors in Chinese queries because those characters out of Chinese characters library cannot be input, the phenomena that they are misused and confused occur frequently [2]. For example, there are some errors, as word omission and addition, reversing order in Chinese. These errors cannot be detected directly. The methods need a big vocabulary base learned from query logs for error checking. But the neologism emerging quickly is rarely provided in the base. For instance, there are 17.4% of queries out of the query vocabulary in the MSN data set. These uncollected vocabularies have greater risks of being misused in queries. The results show that it accounts nearly for 85% in the query error checking. Thus the dynamic refreshed query log is necessary for query correction [3].

In this paper the query logs of SOGOU (http://www.sogou.com/) are used for the model learning and error checking. Although the log formats of different search engines are variations, they have some common data items, such as, query time, user ID, query, and page rank, as Table 1 shows.

The web log contains abundant resources for error checking. We propose a query checking model by mining query log after it has been cleaned by wiping off its noise.

2. Related Work

Query error checking has received great attention these years; their models involve more complex language problems [4, 5]. Large scales of query logs and related corpus are also necessary for their model training. Many machine learning and natural language processing technologies which are very effective in textual entailment, free translation and transliteration, and sentence analysis have been introduced into query checking task [6]. In addition, some correction features are learned from these query logs [7, 8].

There are two technical branches for query error checking, including statistical and rule-based methods, respectively [9, 10]. Rule-based methods use lexical analysis, shallow syntax analysis, or other lingual rules for query analysis and error checking. These methods have higher accuracy

Table 1: Query log format of SOGOU.

Continued time	User ID	Query string	Page rank	Page number	URL
00:00:03	9717831746543397	奥运 (Olympics)	7	3	http://2008.olympic.cn/
00:00:03	7954902679225404	芜湖旅游 (Wuhu Tourism)	7	3	http://www.tourunion.com/spot/city/583340.htm

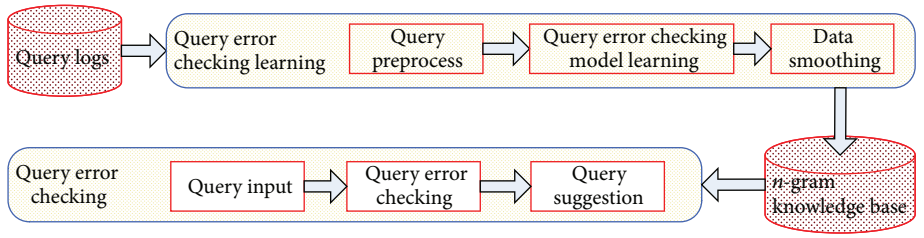

Figure 1: Query error checking model learning and its application.

for query checking. The rules also have limitations for their usage, the query errors are various, and these rules can not apply in all cases. Thus a sufficiently large rule base for query error checking is needed. But the rule base is not easy to acquire. The statistical methods are based on a large corpus by some mathematical algorithms [11]. The frequency of query words can be trained for further use. The frequently cooccurring two words have more possibility of being the meaningful unit; in other words, they prefer to be the correct queries. Thus the interrelationship of the adjacent words is put into the statistical model for query error checking [12]. In this paper we propose the n-gram method with word segmentation information. It combines the language information into statistical model.

3. Query Error Checking Model

3.1. Data Preparing. The queries are put into the input box of search engine. They need some corrections or adjustments to meet user's intents. When some results are returned, the best results will be clicked and also recorded in the log file. All these clues are saved as the query logs. Query log is the user's operation record. These log records are some irregular data because users' operating habits are different. When these logs are used as experimental materials, they must be preprocessed including removing this noise. Finally it provides favorable conditions.

3.2. n-Gram Model for Query Error Checking. We use n-gram training model to train the web log; the training data is used to detect whether the query is correct; if it is not correct query, then it prompts the error checking results; the specific operating process is shown in Figure 1.

In the query error checking model, the most important thing is to calculate the frequency of cooccurrence of words in context. With the help of the prior distribution of words in context, the following word of every query can be predicted by its prior knowledge. This method can be used to check the query. For example, the query "大安门 (it is error form of 'Tiananmen Square'; its correct Chinese form is 天安门)" is an error query that needs query error correction. The error checking method will predict the next word by corpus and choose the cue word for it. The formalization description is as shown below.

Give a query string $s = w_1, w_2, w_3, \ldots, w_n$; s means the query string which is composed of some words w_i. We can compute the prior possibility of this query as

$$\begin{aligned} P(s) &= P(w_1) \times P(w_2 \mid w_1) \times \cdots \\ &\quad \times P(w_n \mid w_1 w_2 \cdots w_{n-1}) \\ &= \prod_i^n P(w_i \mid w_1 w_2 \cdots w_{i-1}), \end{aligned} \tag{1}$$

where w_i represents Chinese word or character; the possibility of w_i depends on its context; here context only means the words ahead of w_i, which is the sequence of $w_1, w_2, \ldots, w_{i-1}$. When $i = 1$, $P(w_1 \mid w_0) = P(w_1)$.

This model depends on context, such as the fact that w_i depends on w_{i-1} or more context words; it is a kind of context. With the increase of the length of context, it will lead to the parameters raising with exponential scale.

If there are L training sets and the length of query strings is n, it will produce L^{i-1} histories for i and L^n free parameters. For example, $L = 6000$ and $i = 3$; the number of free parameters is nearly 216 billion; thus they are difficult to be calculated. The approximate method should be adopted into n-gram model as follows:

$$P(s) = \prod_{i=1}^n P(w_i \mid w_{i-n+1} \cdots w_{i-1}). \tag{2}$$

It is a kind of Markov model for query checking; $P(w_i \mid w_{i-n+1}, \ldots, w_{i-1})$ is its conditional possibility, also as $P(w_i \mid w_{i-n+1}^{i-1})$. Generally speaking, the longer the context window is, the more complex the computing cost is. Thus 2-gram and 3-gram models are often adopted. The Maximum Likelihood

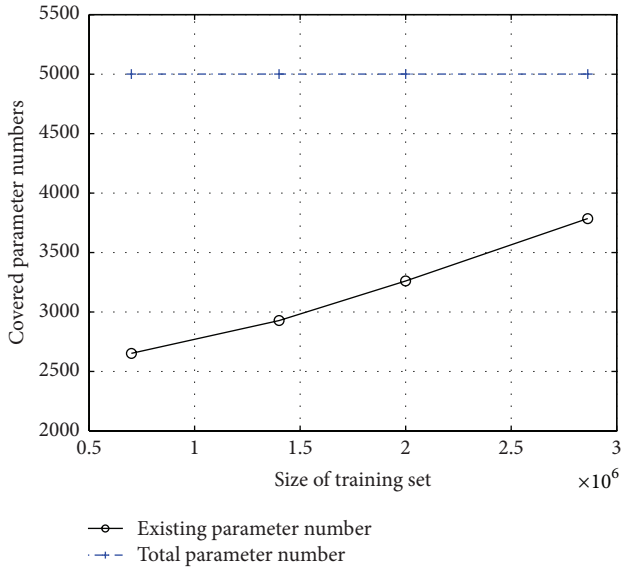

Figure 2: Relation between parameter coverage and size of data set.

Estimation (MLE) is used to estimate the parameters of $P(w_i \mid w_{i-n+1}^{i-1})$ as follows:

$$P\left(w_i \mid w_{i-n+1}^{i-1}\right) = f\left(w_i \mid w_{i-n+1}^{i-1}\right) = \frac{c\left(w_{i-n+1}^{i}\right)}{\sum_{w_i} c\left(w_{i-n+1}^{i-1}\right)}, \quad (3)$$

where $\sum_{w_i} c(w_{i-n+1}^{i})$ is the frequency of query w_{i-n+1}^{i} with w_i cooccurrence in the query log. The following 2-gram model is trained as shown in Figure 2. It is with 5,000 parameters as test set, respectively, with 700,000, 1,400,000, 2,100,000, and 2,800,000 items as the training sets for parameter estimation. Some groups of experiments are designed for parameter estimation. It shows that the larger the training set is, the more reliable the result for parameter estimation is. As the results shown in Figure 2, the label for x-axis is the number of covered parameters and the y-axis is the different training scales.

From Figure 2, we conclude that with the increasing of training scales, the coverage of parameters grows. But there are some words with probabilities of zero in the parameter estimation. We use query logs to train the n-gram model; there are 14.7% queries for trigram and 2.2% queries for bigram without occurring in the training, respectively. The data sparseness problem is triggered by the lack of the training corpus; it will be solved by the data smoothing method in Section 3.3.

3.3. Data Smoothing. The n-gram model needs large training corpus to estimate parameters. When these parameters sometime are not covered, their values will be initialized as zero; it decreases the performance of the algorithm. Thus the data smoothing method is introduced into our model.

There are many kinds of methods for data smoothing. In the experiment, we will use the absolute discounting smoothing operation on the experimental data [13]; its idea is to lower the probability of seen words by subtracting a constant from their counts. It discounts the seen word probability by subtracting a constant. It assumes that $\sum_{w_i} P(w_i \mid w_1, w_2, \ldots, w_{i-1}) = 1$, adjusts the balance between nonzero and zero parameters, and improves the performance of the model as follows:

$$P(s) = \prod_{i=1}^{m+1} P\left(w_i \mid w_{i-n+1}^{i-1}\right). \quad (4)$$

4. Experimental Results

4.1. Data Set. There are about 2,900,000 query logs and removing noise for experiment. After data cleaning, it collects about 440,000 query entries without duplication; its compression ratio is 15.3%.

4.2. Results. According to the proposed method, we use the query log to do the following experiment. Firstly, we choose 10 days of continuous data and label these queries by manual work. Secondly, randomly select three consecutive days' data as the training set and extract 2,000 correct queries and 2,000 error queries, respectively; it consists of 4,000 queries as the test set. Finally segment the queries and train the bigram model with data smoothing processing because the coverage of bigram is good. Then acquire their parameters. We define the Word Cooccurring Distributes (WCD) as (5) when the number of cooccurring word is two, and the other numbers are the same ways as follows:

$$\mathrm{WCD}\left(w_i w_{i+1}\right) = \frac{f\left(w_i w_{i+1}\right)}{\sum_i f\left(w_i w_{i+1}\right)}. \quad (5)$$

Figures 3, 4, 5, and 6 are the distributions when the number of words is 2, 3, 4, and 5 in queries, respectively. For the watching convenience, we use the WCD to describe the correct queries and error queries, respectively, and give them new names as WCD_CQ for correct queries and WCD_EQ for error queries.

From the above figures known, WCD_CQ is above WCD_EQ in certain level. It has relatively clear threshold between correct queries and error queries. The correct queries and error queries can be distinguished when the thresholds are kept in the certain level. Thus this threshold is very significant to distinguish right from error queries. They mean that the correct queries are more frequently used than error queries.

By occasion the error queries are higher than threshold. We check these error queries and find that the error terms are frequently used in the web log. It is the usual thing for most users because they do not concern the spelling sometimes. We can establish the general table for those frequent error queries that are endowed with different threshold.

Through the figures above, we also conclude that under the condition of the same number of query words, in most cases, WCD_CQ tends to be greater than WCD_EQ except the new net words occurring rapidly.

Another phenomenon is that when the number of Chinese characters enlarges, their distribution decreases. Thus

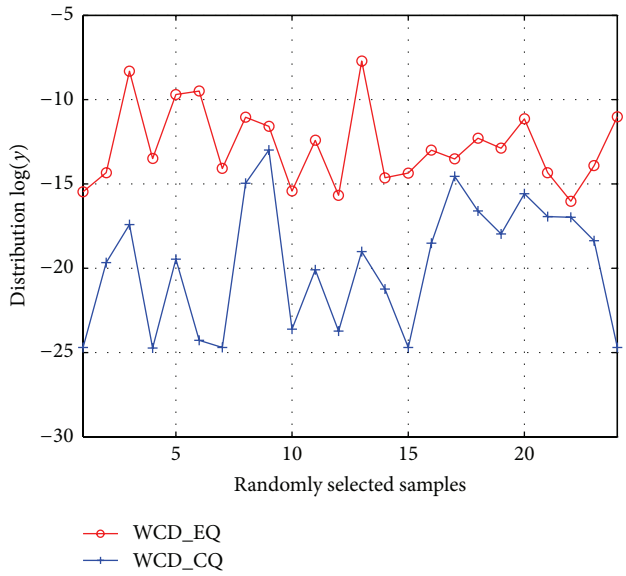

Figure 3: Distribution for two words in queries.

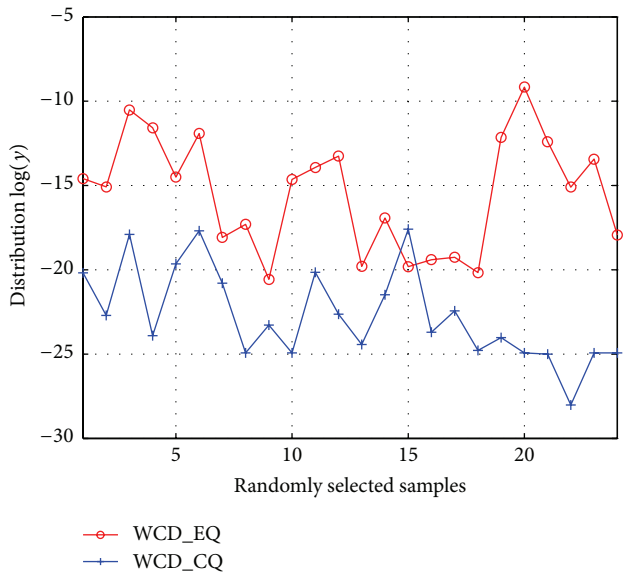

Figure 4: Distribution for three words in queries.

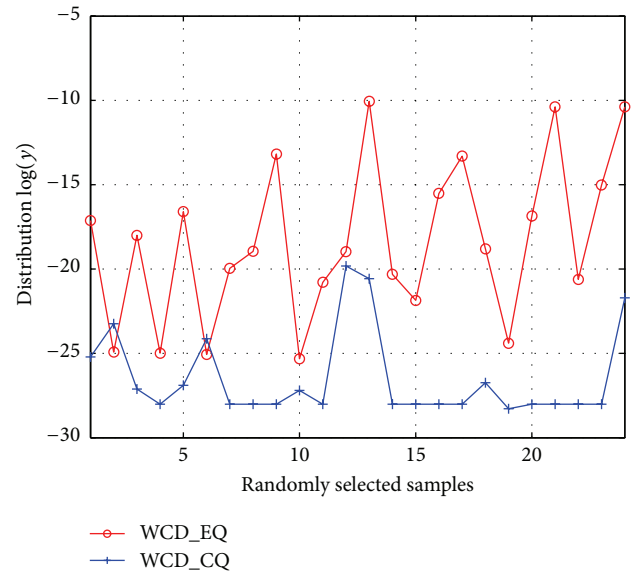

Figure 5: Distribution for four words in queries.

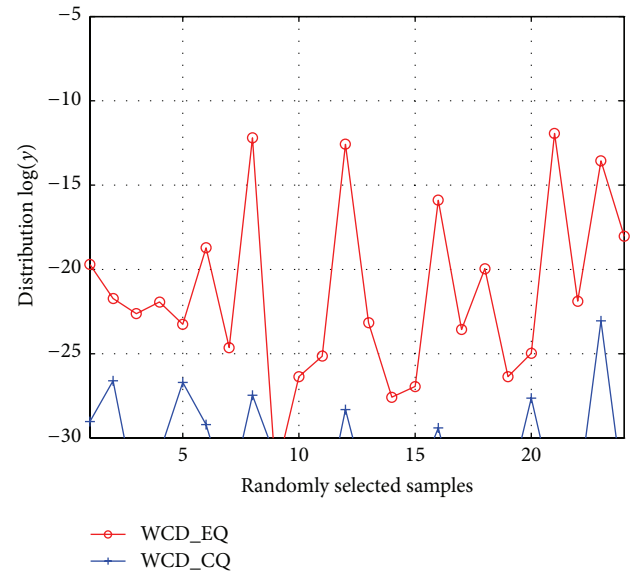

Figure 6: Distribution for five words in queries.

Table 2: Accuracy for correct queries and error queries by Chinese character.

Chinese character number	2	3	4	5
Accuracy of correct queries	84.97%	83.57%	68.39%	60.23%
Accuracy of error queries	89.25%	81.55%	85.57%	95.00%

Table 3: The confused set of measure.

	Judged as right	Judged as wrong
Correct query	A	B
Error query	C	D

we get the relations between number and accuracy as in Table 2.

Here the meaning for measure is shown in Table 3, accuracy of correct queries = $A/(A+B)$, and accuracy of error queries = $D/(C+D)$.

Through the experiments we can draw the following conclusion. The number of Chinese characters in queries has a great influence on a query. When the number increases, its effects on the thresholds will decrease and the ranges of thresholds also gradually turn narrower. It will lead to distinguishing the correct words difficultly.

The results of above several group experiments are consistent with our expected effects. However, with the number increasing, the correct rate and the discrimination of this feature will drop down. It needs further investigation.

Table 4: Improved results by word between correct queries and error queries.

Word number	2	3	4	5
Correct queries	95.95%	85.92%	78.16%	71.59%
Error queries	86.02%	79.13%	84.08%	84.17%

Besides the number of Chinese characters and its possibility of affecting the error checking between correct and error queries, the number of Chinese word is also a kind of important feature. We analyze the correct queries and wrong queries, respectively as shown in Table 4; the number of Chinese words in correct queries is longer than that in wrong queries. Thus it means that it is an important feature for query error checking.

When the number of Chinese word as a feature is added into the model, the results are improved significantly. Table 4 shows that the accuracy rate of correct queries increases to be higher than that of error queries. It means that this feature is effective.

5. Conclusion

In this paper, we propose an error checking model n-gram model. The error checking method uses the different Chinese character number of queries as a kind of feature and gets their threshold to check the query. Then the Chinese word number of queries is introduced into the model to improve the performance. The new feature combination increases the accuracy of correct queries and the recall of error queries.

Although this method achieves the anticipated effect, when the word number of queries is more 6, its performance will decrease. The following work will continue to improve the error checking method.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

The authors are grateful to the reviewers for reviewing this paper. This work is supported by the National Science Foundation of China (Grant no. 61103112), Social Science Foundation of Beijing (Grant no. 13SHC031), Ministry of Education of China (Project of Humanities and Social Sciences, Grant no. 13YJC740055), and Beijing Young Talent Plan (Grant no. CIT&TCD201404005).

References

[1] F. Kaveh-Yazdy and A.-M. Zareh-Bidoki, "Aleph or Aleph-Maddah, that is the question! Spelling correction for search engine autocomplete service," in *Proceedings of the 4th International Conference on Computer and Knowledge Engineering (ICCKE '14)*, pp. 273–278, October 2014.

[2] P. Jin, X. Chen, and Z. Guo, "Integrating pinyin to improve spelling errors detection for Chinese language," in *Proceedings of the IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies*, pp. 455–458, 2014.

[3] M. S. Rasooli, O. Kahefi, and B. Minaei-Bidgoli, "Effect of adaptive spell checking in Persian," in *Proceedings of the 7th International Conference on Natural Language Processing and Knowledge Engineering (NLP-KE '11)*, pp. 161–164, November 2011.

[4] J. Li and X. Wang, "Combining trigram and automatic weight distribution in Chinese spelling error correction," *Journal of Computer Science and Technology*, vol. 17, no. 6, pp. 915–923, 2002.

[5] R.-Y. Chang, C.-H. Wu, and P. K. Prasetyo, "Error diagnosis of chinese sentences using inductive learning algorithm and decomposition-based testing mechanism," *ACM Transactions on Asian Language Information Processing*, vol. 11, no. 1, article 3, 2012.

[6] M. Kim, J. Jin, H.-C. Kwon, and A. Yoon, "Statistical context-sensitive spelling correction using typing error rate," in *Proceedings of the 16th IEEE International Conference on Computational Science and Engineering (CSE '13)*, pp. 1242–1246, IEEE, Sydney, Australia, December 2013.

[7] M. Kim, S.-K. Choi, and H.-C. Kwon, "Context-sensitive spelling error correction using inter-word semantic relation analysis," in *Proceedings of the 5th International Conference on Information Science and Applications (ICISA '14)*, May 2014.

[8] B. Siklosi, A. Novak, and G. Proszeky, "Context-aware correction of spelling errors in Hungarian medical documents," in *Statistical Language and Speech Processing*, vol. 7978 of *Lecture Notes in Computer Science*, pp. 248–259, Springer, Berlin, Germany, 2013.

[9] M. Y. Soleh and A. Purwarianti, "A non word error spell checker for Indonesian using morphologically analyzer and HMM," in *Proceedings of the International Conference on Electrical Engineering and Informatics (ICEEI '11)*, pp. 1–8, July 2011.

[10] S. Sulaiman, K. Omar, N. Omar, M. Z. Murah, and H. A. Rahman, "Spelling error detector rule for Jawi stemmer," in *Proceedings of the International Conference on Pattern Analysis and Intelligent Robotics (ICPAIR '11)*, pp. 78–82, June 2011.

[11] G. Dalkiliç and Y. Çebi, "Turkish spelling error detection and correction by using word N-grams," in *Proceedings of the 5th International Conference on Soft Computing, Computing with Words and Perceptions in System Analysis, Decision and Control (ICSCCW '09)*, September 2009.

[12] V. J. Hodge and J. Austin, "A comparison of standard spell checking algorithms and a novel binary neural approach," *IEEE Transactions on Knowledge and Data Engineering*, vol. 15, no. 5, pp. 1073–1081, 2003.

[13] H. Ney, U. Essen, and R. Kneser, "On the estimation of 'small' probabilities by leaving-one-out," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 12, pp. 1202–1212, 1995.