Perceptron-based Tagging of Query Boundaries for Chinese Query Segmentation

Jingfei Du[1,2][*] Yan Song[1] Chi-Ho Li[1]
[1]Microsoft Search Technology Center Asia
5# Danling St., Haidian District, Beijing, China
[2]Beijing University of Posts and Telecommunications
10# Xi Tu Cheng St., Haidian District, Beijing, China
jingfeidu@gmail.com; {yansong, chl}@microsoft.com

ABSTRACT

Query boundaries carry useful information for query segmentation, especially when analyzing queries in a language with no space, e.g., Chinese. This paper presents our research on Chinese query segmentation via averaged perceptron to model query boundaries through an *L*-*R* tagging scheme on a large amount of unlabeled queries. Experimental results indicate that query boundaries are very informative and they significantly improve supervised Chinese query segmentation when labeled training data is very limited.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Retrieval Models

Keywords

Query Segmentation, Query Boundary, Averaged Perceptron

1. INTRODUCTION

Query segmentation is a necessary and important initial step in a search engine. It is difficult and especially challenging for languages like Chinese, where there is no space as word boundaries. A straightforward solution to boundary identification is to apply word segmentation technique in natural language processing (NLP), where there are many state-of-the-art models can be trained on human annotated corpora such as the Penn Chinese Treebank (CTB)[1]. However, query segmentation cannot be directly performed by conventional word segmentation methods because web search query language is very different, keywords in user queries are usually not the same as words in a natural language corpus (see results in Section 3). In addition, there is not enough labeled query data for query segmentation. Many researches have been done with unsupervised approach to query segmentation or with additional resources [2, 4] [2].

诺	基	亚	手	机	性	能	价	格
L1	L0	L0	L0	L0	L1	L0	L1	L0
R0	R0	R0	R0	R1	R0	R1	R0	R1

Table 1: *L*-*R* Labels for a Query Sample.

In order to build a query segmentation model with limited labeled data, we propose to use boundary information from query logs to help query segmentation. The method is based on the assumption that query boundaries can be seen as natural annotations provided by users. We adopt a well performed labeling method to generate training data for query segmentation from query boundaries, and build unsupervised and semi-supervised models on the data. As we shall see, experimental results show that query boundary information learned from a large collection of queries is an effective guidance for query segmentation, especially when there are only a small amount of labeled queries available.

2. METHOD

2.1 Query Boundary Labeling

The key of query segmentation is to find the boundaries of units in a query, where a unit in a Chinese query is either a single word or a phrase. We use s *L*-*R* tagging scheme to label segmentation boundary information in a query, where the boundary of a unit is represented by its left-most (L) and right-most (R) characters, as shown in Table 1. For L, a binary classification model can be used to decide whether a character should be assign with a positive boundary tag *L1* or a negative boundary tag *L0*. The same is true for R model. In addition, the *L*-*R* tagging scheme can be transformed into a widely used tagging scheme in Chinese word segmentation [3], wherein each character is assigned one of the position tags: B(beginning), I(middle), E(ending) and S(single). The transform approach is shown in Eq. 1. This

[*] This work was done during the first author's internship at Microsoft Search Technology Center Asia.

[1] http://www.cis.upenn.edu/ chinese/ctb.html

[2] Owing to that they used additional resources and were designed generically for all languages, therefore unable to address the problem in Chinese. We will not compare them directly with our methods in this paper.

WWW'14 Companion, April 7–11, 2014, Seoul, Korea.
ACM 978-1-4503-2745-9/14/04.
http://dx.doi.org/10.1145/2567948.2577377 .

fact helps to build unsupervised segmentation model.

$$B \leftarrow L1R0 \quad I \leftarrow L0R0 \quad E \leftarrow L0R1 \quad S \leftarrow L1R1 \tag{1}$$

2.2 Segmentation Models

We use averaged perceptron [1] to train our models, as shown in Algorithm 1.

Algorithm 1 Averaged Perceptron

```
Input: Training data $C$, Iteration number $T$
$\vec{\omega} \leftarrow 0, \vec{\omega}_a \leftarrow 0, c \leftarrow 1$
for $t = 1$ to $T$ do
    for $(x_i, y_i) \in C$ do
        $\hat{y}_i \leftarrow argmax_{y_i} \vec{\omega} \cdot \Phi(x_i, y_i)$
        if $\hat{y}_i \neq y_i^*$ then
            $\vec{\omega} \leftarrow \vec{\omega} + \Phi(x_i^*, y_i^*) - \Phi(\hat{x}_i, \hat{y}_i)$
            $\vec{\omega}_a \leftarrow \vec{\omega}_a + c \cdot \{\Phi(x_i^*, y_i^*) - \Phi(\hat{x}_i, \hat{y}_i)\}$
        $c \leftarrow c + 1$
return $\vec{\omega} - \vec{\omega}_a / c$
```

In our experiments, the perceptron algorithm is applied to a variety of training corpora, including the CTB 5.0, the raw query log, and the query log with manual segmentation. It is also tried with both the *BIES* and *L-R* tagging scheme. The features to be used always include a set of baseline features [3] , and in some settings there are also some additional features. Such additional features can be the tags produced by another instance of the perceptron algorithm that uses some other training corpus, or can be the features produced by an unsupervised method known as DLG [3]. The DLG method generates a lot of n-gram features from raw queries, and some of these features can be filtered if the corresponding n-grams do not match the *L-R* tags produced by another instance of the perceptron algorithm.

3. EVALUATION

We evaluate our methods on query logs from Bing China (http://cn.bing.com). Around 25M[3] queries are collected from 09-18-2013 to 09-24-2014 in order to test our approach. Most of the queries are in Chinese. We manually annotated 100K queries as test data and 10K for training supervised models[4]. All queries except the test data are used to learn boundary information. Due to the limit of space, here query segmentation is evaluated as if it is a stand-alone task, and F-score[5] is used as the evaluation metric in this paper. The results of our experiments are tested via a two-tailed 10-chunk partitioned t-test to determine whether the differences compared with #4 are statistically significant.

The results are presented in Table 2. It can be obviously seen that the conventional Chinese word segmentation approach, viz. *BIES* tagging based on CTB (#1), gives very poor result, and its improvement by adding training data of annotated queries (#2) is still limited. However, if the *L-R* tagging scheme is adopted, then even raw queries as training data (#3) leads to a much better performance. Such performance can be boosted if annotated queries are taken as training data (#4). The fact confirms that a model trained from a conventional natural language corpus can barely help query segmentation because such corpus is very different from queries. In addition, query boundaries did help query segmentation, as even a simple *L-R* model built on raw queries can achieve nearly 0.6 F-score.

Id	Scheme	Data	Features	F score
#1	BIES	C	baseline	0.3077
#2	BIES	C+A	baseline	0.4470
#3	LR-BIES	R	baseline	0.5822
#4	BIES	A	baseline	0.6131
#5	LR+BIES	A	+ tags from #3	0.6144
#6	LR+BIES	A	+ DLG features	0.6582*
#7	LR+BIES	A	+ DLG filtered by #3 tags	0.8662**

Table 2: Performance comparison of different methods. * and ** indicate t-test significance level at 0.05 and 0.01 over the result in #4. A, C, R refer to annotated queries, CTB 5.0 corpus and raw queries, respectively. LR-BIES refers to that *L-R* tags are transformed to *BIES* tags, LR+BIES refers to that both *LR* and *BIES* tagging schemes are used.

As to the impact of feature modeling, adding the *L-R* tags from #3 as additional features in #4 (i.e. #5 [6]) does not help much. Neither does the vast amount of n-gram features by DLG (#6) bring huge improvement. Note that DLG features are extracted from all raw queries. Yet if we combine both of them (#7), viz. to filter the DLG n-gram features by the *L-R* tags from #3, the final F-score is boosted for more than 90% relatively. Moreover, this method also discards more than 90% of the n-grams in #6, thereby leading to a much faster convergence of the training process.

4. CONCLUSIONS

We proposed to use unlabeled query data to train an unsupervised segmentation model and use it to refine n-gram features used in a semi-supervised model. The results indicate that boundary information learned from a large number of queries can effectively help Chinese query segmentation. In future research we would like to test our segmentation approach in an end-to-end environment and measure how it improves search accuracy and relevance given Chinese queries.

5. REFERENCES

[1] M. Collins. Discriminative Training Methods for Hidden Markov Models: Theory and Experiments with Perceptron Algorithms. In *EMNLP*, pages 1–8, 2002.

[2] N. Mishra, R. Saha Roy, N. Ganguly, S. Laxman, and M. Choudhury. Unsupervised Query Segmentation Using only Query Logs. In *WWW*, pages 91–92, 2011.

[3] Y. Song and F. Xia. Using a Goodness Measurement for Domain Adaptation: A Case Study on Chinese Word Segmentation. In *LREC*, pages 3853–3860, 2012.

[4] B. Tan and F. Peng. Unsupervised Query Segmentation using Generative Language Models and Wikipedia. In *WWW*, pages 347–356, 2008.

[3] Here M (million) and K (thousand) refer to query number.

[4] We will try to have this data set available.

[5] $F = 2PR/(P + R)$, where P and R refer to precision and recall, respectively.

[6] For LR+BIES, *BIES* is used to train on labeled query data and *LR* is for raw queries.