Phrase-Based Translation Model for Question Retrieval in Community Question Answer Archives

Guangyou Zhou, Li Cai, Jun Zhao*, and Kang Liu
National Laboratory of Pattern Recognition
Institute of Automation, Chinese Academy of Sciences
95 Zhongguancun East Road, Beijing 100190, China
{gyzhou,lcai,jzhao,kliu}@nlpr.ia.ac.cn

Abstract

Community-based question answer (Q&A) has become an important issue due to the popularity of Q&A archives on the web. This paper is concerned with the problem of question retrieval. Question retrieval in Q&A archives aims to find historical questions that are semantically equivalent or relevant to the queried questions. In this paper, we propose a novel phrase-based translation model for question retrieval. Compared to the traditional word-based translation models, the phrase-based translation model is more effective because it captures contextual information in modeling the translation of phrases as a whole, rather than translating single words in isolation. Experiments conducted on real Q&A data demonstrate that our proposed phrase-based translation model significantly outperforms the state-of-the-art word-based translation model.

1 Introduction

Over the past few years, large scale question and answer (Q&A) archives have become an important information resource on the Web. These include the traditional Frequently Asked Questions (FAQ) archives and the emerging community-based Q&A services, such as Yahoo! Answers[1], Live QnA[2], and Baidu Zhidao[3].

*Correspondence author: jzhao@nlpr.ia.ac.cn

[1]http://answers.yahoo.com/

[2]http://qna.live.com/

[3]http://zhidao.baidu.com/

Community-based Q&A services can directly return answers to the queried questions instead of a list of relevant documents, thus provide an effective alternative to the traditional adhoc information retrieval. To make full use of the large scale archives of question-answer pairs, it is critical to have functionality helping users to retrieve historical answers (Duan et al., 2008). Therefore, it is a meaningful task to retrieve the questions that are semantically equivalent or relevant to the queried questions. For example in Table 1, given question Q_1, Q_2 can be returned and their answers will then be used to answer Q_1 because the answer of Q_2 is expected to partially satisfy the queried question Q_1. This is what we called *question retrieval* in this paper.

The major challenge for Q&A retrieval, as for most information retrieval models, such as vector space model (VSM) (Salton et al., 1975), Okapi model (Robertson et al., 1994), language model (LM) (Ponte and Croft, 1998), is the *lexical gap* (or *lexical chasm*) between the queried questions and the historical questions in the archives (Jeon et al., 2005; Xue et al., 2008). For example in Table 1, Q_1 and Q_2 are two semantically similar questions, but they have very few words in common. This prob-

Query:
Q_1: How to get rid of **stuffy nose**?
Expected:
Q_2: What is the best way to prevent a cold?
Not Expected:
Q_3: How do I air out my stuffy room?
Q_4: How do you make a nose bleed stop quicker?

Table 1: An example on question retrieval

Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics, pages 653–662,
Portland, Oregon, June 19-24, 2011.

lem is more serious for Q&A retrieval, since the question-answer pairs are usually short and there is little chance of finding the same content expressed using different wording (Xue et al., 2008). To solve the lexical gap problem, most researchers regarded the question retrieval task as a statistical machine translation problem by using IBM model 1 (Brown et al., 1993) to learn the word-to-word translation probabilities (Berger and Lafferty, 1999; Jeon et al., 2005; Xue et al., 2008; Lee et al., 2008; Bernhard and Gurevych, 2009). Experiments consistently reported that the word-based translation models could yield better performance than the traditional methods (e.g., VSM. Okapi and LM). However, all these existing approaches are considered to be *context independent* in that they do not take into account any contextual information in modeling word translation probabilities. For example in Table 1, although neither of the individual word pair (e.g., "stuffy"/"cold" and "nose"/"cold") might have a high translation probability, the sequence of words "stuffy nose" can be easily translated from a single word "cold" in Q_2 with a relative high translation probability.

In this paper, we argue that it is beneficial to capture contextual information for question retrieval. To this end, inspired by the phrase-based statistical machine translation (SMT) systems (Koehn et al., 2003; Och and Ney, 2004), we propose a phrase-based translation model (P-Trans) for question retrieval, and we assume that question retrieval should be performed at the phrase level. This model learns the probability of translating one sequence of words (e.g., phrase) into another sequence of words, e.g., translating a phrase in a historical question into another phrase in a queried question. Compared to the traditional word-based translation models that account for translating single words in isolation, the phrase-based translation model is potentially more effective because it captures some contextual information in modeling the translation of phrases as a whole. More precise translation can be determined for phrases than for words. It is thus reasonable to expect that using such phrase translation probabilities as ranking features is likely to improve the question retrieval performance, as we will show in our experiments.

Unlike the general natural language translation, the parallel sentences between questions and answers in community-based Q&A have very different lengths, leaving many words in answers unaligned to any word in queried questions. Following (Berger and Lafferty, 1999), we restrict our attention to those phrase translations consistent with a good word-level alignment.

Specifically, we make the following contributions:

- we formulate the question retrieval task as a phrase-based translation problem by modeling the contextual information (in Section 3.1).
- we linearly combine the phrase-based translation model for the question part and answer part (in Section 3.2).
- we propose a linear ranking model framework for question retrieval in which different models are incorporated as features because the phrase-based translation model cannot be interpolated with a unigram language model (in Section 3.3).
- finally, we conduct the experiments on community-based Q&A data for question retrieval. The results show that our proposed approach significantly outperforms the baseline methods (in Section 4).

The remainder of this paper is organized as follows. Section 2 introduces the existing state-of-the-art methods. Section 3 describes our phrase-based translation model for question retrieval. Section 4 presents the experimental results. In Section 5, we conclude with ideas for future research.

2 Preliminaries

2.1 Language Model

The unigram language model has been widely used for question retrieval on community-based Q&A data (Jeon et al., 2005; Xue et al., 2008; Cao et al., 2010). To avoid zero probability, we use Jelinek-Mercer smoothing (Zhai and Lafferty, 2001) due to its good performance and cheap computational cost. So the ranking function for the query likelihood language model with Jelinek-Mercer smoothing can be

written as:

$$Score(\mathbf{q}, D) = \prod_{w \in \mathbf{q}} (1-\lambda) P_{ml}(w|D) + \lambda P_{ml}(w|C) \quad (1)$$

$$P_{ml}(w|D) = \frac{\#(w, D)}{|D|}, \;\; P_{ml}(w|C) = \frac{\#(w, C)}{|C|} \quad (2)$$

where $\mathbf{q}$ is the queried question, D is a document, C is background collection, λ is smoothing parameter. $\#(t, D)$ is the frequency of term t in D, $|D|$ and $|C|$ denote the length of D and C respectively.

2.2 Word-Based Translation Model

Previous work (Berger et al., 2000; Jeon et al., 2005; Xue et al., 2008) consistently reported that the word-based translation models (Trans) yielded better performance than the traditional methods (VSM, Okapi and LM) for question retrieval. These models exploit the word translation probabilities in a language modeling framework. Following Jeon et al. (2005) and Xue et al. (2008), the ranking function can be written as:

$$Score(\mathbf{q}, D) = \prod_{w \in \mathbf{q}} (1-\lambda) P_{tr}(w|D) + \lambda P_{ml}(w|C) \quad (3)$$

$$P_{tr}(w|D) = \sum_{t \in D} P(w|t) P_{ml}(t|D), \;\; P_{ml}(t|D) = \frac{\#(t, D)}{|D|} \quad (4)$$

where $P(w|t)$ denotes the translation probability from word t to word w.

2.3 Word-Based Translation Language Model

Xue et al. (2008) proposed to linearly mix two different estimations by combining language model and word-based translation model into a unified framework, called TransLM. The experiments show that this model gains better performance than both the language model and the word-based translation model. Following Xue et al. (2008), this model can be written as:

$$Score(\mathbf{q}, D) = \prod_{w \in \mathbf{q}} (1-\lambda) P_{mx}(w|D) + \lambda P_{ml}(w|C) \quad (5)$$

$$P_{mx}(w|D) = \alpha \sum_{t \in D} P(w|t) P_{ml}(t|D) + (1-\alpha) P_{ml}(w|D) \quad (6)$$

D:	… for good cold home remedies …	*document*
E:	[for, good, cold, home remedies]	*segmentation*
F:	[for$_1$, best$_2$, stuffy nose$_3$, home remedy$_4$]	*translation*
M:	(1→3, 2→1, 3→4, 4→2)	*permutation*
$\mathbf{q}$:	best home remedy for stuffy nose	*queried question*

Figure 1: Example describing the generative procedure of the phrase-based translation model.

3 Our Approach: Phrase-Based Translation Model for Question Retrieval

3.1 Phrase-Based Translation Model

Phrase-based machine translation models (Koehn et al., 2003; D. Chiang, 2005; Och and Ney, 2004) have shown superior performance compared to word-based translation models. In this paper, the goal of phrase-based translation model is to translate a document[4] D into a queried question $\mathbf{q}$. Rather than translating single words in isolation, the phrase-based model translates one sequence of words into another sequence of words, thus incorporating contextual information. For example, we might learn that the phrase "stuffy nose" can be translated from "cold" with relative high probability, even though neither of the individual word pairs (e.g., "stuffy"/"cold" and "nose"/"cold") might have a high word translation probability. Inspired by the work of (Sun et al., 2010; Gao et al., 2010), we assume the following generative process: first the document D is broken into K non-empty word sequences $\mathbf{t}_1, \ldots, \mathbf{t}_K$, then each $\mathbf{t}$ is translated into a new non-empty word sequence $\mathbf{w}_1, \ldots, \mathbf{w}_K$, and finally these phrases are permutated and concatenated to form the queried questions $\mathbf{q}$, where $\mathbf{t}$ and $\mathbf{w}$ denote the phrases or consecutive sequence of words.

To formulate this generative process, let E denote the segmentation of D into K phrases $\mathbf{t}_1, \ldots, \mathbf{t}_K$, and let F denote the K translation phrases $\mathbf{w}_1, \ldots, \mathbf{w}_K$ —we refer to these $(\mathbf{t}_i, \mathbf{w}_i)$ pairs as *bi-phrases*. Finally, let M denote a permutation of K elements representing the final reordering step. Figure 1 describes an example of the generative procedure.

Next let us place a probability distribution over rewrite pairs. Let $B(D, \mathbf{q})$ denote the set of E,

[4] In this paper, a document has the same meaning as a historical question-answer pair in the Q&A archives.

F, M triples that translate D into $\mathbf{q}$. Here we assume a uniform probability over segmentations, so the phrase-based translation model can be formulated as:

$$P(\mathbf{q}|D) \propto \sum_{\substack{(E,F,M)\in \\ B(D,\mathbf{q})}} P(F|D,E) \cdot P(M|D,E,F) \quad (7)$$

As is common practice in SMT, we use the maximum approximation to the sum:

$$P(\mathbf{q}|D) \approx \max_{\substack{(E,F,M)\in \\ B(D,\mathbf{q})}} P(F|D,E) \cdot P(M|D,E,F) \quad (8)$$

Although we have defined a generative model for translating D into $\mathbf{q}$, our goal is to calculate the ranking score function over existing $\mathbf{q}$ and D, rather than generating new queried questions. Equation (8) cannot be used directly for document ranking because $\mathbf{q}$ and D are often of very different lengths, leaving many words in D unaligned to any word in $\mathbf{q}$. This is the key difference between the community-based question retrieval and the general natural language translation. As pointed out by Berger and Lafferty (1999) and Gao et al. (2010), document-query translation requires a *distillation* of the document, while translation of natural language tolerates little being thrown away.

Thus we attempt to extract the *key document words* that form the distillation of the document, and assume that a queried question is translated only from the *key document words*. In this paper, the key document words are identified via word alignment. We introduce the "hidden alignments" $A = a_1 \dots a_j \dots a_J$, which describe the mapping from a word position j in queried question to a document word position $i = a_j$. The different alignment models we present provide different decompositions of $P(\mathbf{q}, A|D)$. We assume that the position of the *key document words* are determined by the Viterbi alignment, which can be obtained using IBM model 1 as follows:

$$\begin{aligned}\hat{A} &= \arg\max_{A} P(\mathbf{q}, A|D) \\ &= \arg\max_{A} \Big\{ P(J|I) \prod_{j=1}^{J} P(w_j|t_{a_j}) \Big\} \\ &= \Big[\arg\max_{a_j} P(w_j|t_{a_j}) \Big]_{j=1}^{J} \end{aligned} \quad (9)$$

Given $\hat{A}$, when scoring a given Q&A pair, we restrict our attention to those E, F, M triples that are consistent with $\hat{A}$, which we denote as $B(D, \mathbf{q}, \hat{A})$. Here, consistency requires that if two words are aligned in $\hat{A}$, then they must appear in the same bi-phrase $(\mathbf{t}_i, \mathbf{w}_i)$. Once the word alignment is fixed, the final permutation is uniquely determined, so we can safely discard that factor. Thus equation (8) can be written as:

$$P(\mathbf{q}|D) \approx \max_{(E,F,M)\in B(D,\mathbf{q},\hat{A})} P(F|D,E) \quad (10)$$

For the sole remaining factor $P(F|D,E)$, we make the assumption that a segmented queried question $F = \mathbf{w}_1, \dots, \mathbf{w}_K$ is generated from left to right by translating each phrase $\mathbf{t}_1, \dots, \mathbf{t}_K$ independently:

$$P(F|D,E) = \prod_{k=1}^{K} P(\mathbf{w}_k|\mathbf{t}_k) \quad (11)$$

where $P(\mathbf{w}_k|\mathbf{t}_k)$ is a phrase translation probability, the estimation will be described in Section 3.3.

To find the maximum probability assignment efficiently, we use a dynamic programming approach, somewhat similar to the monotone decoding algorithm described in (Och, 2002). We define α_j to be the probability of the most likely sequence of phrases covering the first j words in a queried question, then the probability can be calculated using the following recursion:

(1) Initialization:

$$\alpha_0 = 1 \quad (12)$$

(2) Induction:

$$\alpha_j = \sum_{j'<j, \mathbf{w}=w_{j'+1}\dots w_j} \Big\{ \alpha_{j'} P(\mathbf{w}|\mathbf{t}_{\mathbf{w}}) \Big\} \quad (13)$$

(3) Total:

$$P(\mathbf{q}|D) = \alpha_J \quad (14)$$

3.2 Phrase-Based Translation Model for Question Part and Answer Part

In Q&A, a document D is decomposed into $(\bar{\mathbf{q}}, \bar{\mathbf{a}})$, where $\bar{\mathbf{q}}$ denotes the question part of the historical question in the archives and $\bar{\mathbf{a}}$ denotes the answer part. Although it has been shown that doing Q&A retrieval based solely on the answer part does not perform well (Jeon et al., 2005; Xue et al., 2008), the answer part should provide additional evidence about relevance and, therefore, it should be combined with the estimation based on the question part.

In this combined model, $P(\mathbf{q}|\bar{\mathbf{q}})$ and $P(\mathbf{q}|\bar{\mathbf{a}})$ are calculated with equations (12) to (14). So $P(\mathbf{q}|D)$ will be written as:

$$P(\mathbf{q}|D) = \mu_1 P(\mathbf{q}|\bar{\mathbf{q}}) + \mu_2 P(\mathbf{q}|\bar{\mathbf{a}}) \quad (15)$$

where $\mu_1 + \mu_2 = 1$.

In equation (15), the relative importance of question part and answer part is adjusted through μ_1 and μ_2. When $\mu_1 = 1$, the retrieval model is based on phrase-based translation model for the question part. When $\mu_2 = 1$, the retrieval model is based on phrase-based translation model for the answer part.

3.3 Parameter Estimation

3.3.1 Parallel Corpus Collection

In Q&A archives, question-answer pairs can be considered as a type of parallel corpus, which is used for estimating the translation probabilities. Unlike the bilingual machine translation, the questions and answers in a Q&A archive are written in the same language, the translation probability can be calculated through setting either as the source and the other as the target. In this paper, $P(\bar{\mathbf{a}}|\bar{\mathbf{q}})$ is used to denote the translation probability with the question as the source and the answer as the target. $P(\bar{\mathbf{q}}|\bar{\mathbf{a}})$ is used to denote the opposite configuration.

For a given word or phrase, the related words or phrases differ when it appears in the question or in the answer. Following Xue et al. (2008), a pooling strategy is adopted. First, we pool the question-answer pairs used to learn $P(\bar{\mathbf{a}}|\bar{\mathbf{q}})$ and the answer-question pairs used to learn $P(\bar{\mathbf{q}}|\bar{\mathbf{a}})$, and then use IBM model 1 (Brown et al., 1993) to learn the combined translation probabilities. Suppose we use the collection $\{(\bar{\mathbf{q}}, \bar{\mathbf{a}})_1, \ldots, (\bar{\mathbf{q}}, \bar{\mathbf{a}})_m\}$ to learn $P(\bar{\mathbf{a}}|\bar{\mathbf{q}})$ and use the collection $\{(\bar{\mathbf{a}}, \bar{\mathbf{q}})_1, \ldots, (\bar{\mathbf{a}}, \bar{\mathbf{q}})_m\}$ to learn $P(\bar{\mathbf{q}}|\bar{\mathbf{a}})$, then $\{(\bar{\mathbf{q}}, \bar{\mathbf{a}})_1, \ldots, (\bar{\mathbf{q}}, \bar{\mathbf{a}})_m, (\bar{\mathbf{a}}, \bar{\mathbf{q}})_1, \ldots, (\bar{\mathbf{a}}, \bar{\mathbf{q}})_m\}$ is used here to learn the combination translation probability $P_{pool}(w_i|t_j)$.

3.3.2 Parallel Corpus Preprocessing

Unlike the bilingual parallel corpus used in SMT, our parallel corpus is collected from Q&A archives, which is more noisy. Directly using the IBM model 1 can be problematic, it is possible for translation model to contain "unnecessary" translations (Lee et al., 2008). In this paper, we adopt a variant of TextRank algorithm (Mihalcea and Tarau, 2004) to identify and eliminate unimportant words from parallel corpus, assuming that a word in a question or answer is unimportant if it holds a relatively low significance in the parallel corpus.

Following (Lee et al., 2008), the ranking algorithm proceeds as follows. First, all the words in a given document are added as vertices in a graph G. Then edges are added between words if the words co-occur in a fixed-sized window. The number of co-occurrences becomes the weight of an edge. When the graph is constructed, the score of each vertex is initialized as 1, and the PageRank-based ranking algorithm is run on the graph iteratively until convergence. The TextRank score of a word w in document D at kth iteration is defined as follows:

$$R^k_{w,D} = (1-d) + d \cdot \sum_{\forall j:(i,j)\in G} \frac{e_{i,j}}{\sum_{\forall l:(j,l)\in G} e_{j,l}} R^{k-1}_{w,D} \quad (16)$$

where d is a damping factor usually set to 0.85, and $e_{i,j}$ is an edge weight between i and j.

We use average *TextRank score* as threshold: words are removed if their scores are lower than the average score of all words in a document.

3.3.3 Translation Probability Estimation

After preprocessing the parallel corpus, we will calculate $P(\mathbf{w}|\mathbf{t})$, following the method commonly used in SMT (Koehn et al., 2003; Och, 2002) to extract bi-phrases and estimate their translation probabilities.

First, we learn the word-to-word translation probability using IBM model 1 (Brown et al., 1993). Then, we perform Viterbi word alignment according to equation (9). Finally, the bi-phrases that are consistent with the word alignment are extracted using the heuristics proposed in (Och, 2002). We set the maximum phrase length to five in our experiments.

After gathering all such bi-phrases from the training data, we can estimate conditional relative frequency estimates without smoothing:

$$P(\mathbf{w}|\mathbf{t}) = \frac{N(\mathbf{t}, \mathbf{w})}{N(\mathbf{t})} \quad (17)$$

where $N(\mathbf{t}, \mathbf{w})$ is the number of times that $\mathbf{t}$ is aligned to $\mathbf{w}$ in training data. These estimates are

source	stuffy nose	internet explorer
1	stuffy nose	internet explorer
2	cold	ie
3	stuffy	internet browser
4	sore throat	explorer
5	sneeze	browser

Table 2: Phrase translation probability examples. Each column shows the top 5 target phrases learned from the word-aligned question-answer pairs.

useful for contextual lexical selection with sufficient training data, but can be subject to data sparsity issues (Sun et al., 2010; Gao et al., 2010). An alternate translation probability estimate not subject to data sparsity is the so-called *lexical weight* estimate (Koehn et al., 2003). Let $P(w|t)$ be the word-to-word translation probability, and let A be the word alignment between $\mathbf{w}$ and $\mathbf{t}$. Here, the word alignment contains (i, j) pairs, where $i \in 1 \ldots |\mathbf{w}|$ and $j \in 0 \ldots |\mathbf{t}|$, with 0 indicating a null word. Then we use the following estimate:

$$P_t(\mathbf{w}|\mathbf{t}, A) = \prod_{i=1}^{|\mathbf{w}|} \frac{1}{|\{j|(j,i) \in A\}|} \sum_{\forall (i,j) \in A} P(w_i|t_j) \tag{18}$$

We assume that for each position in $\mathbf{w}$, there is either a single alignment to 0, or multiple alignments to non-zero positions in $\mathbf{t}$. In fact, equation (18) computes a product of per-word translation scores; the per-word scores are the averages of all the translations for the alignment links of that word. The word translation probabilities are calculated using IBM 1, which has been widely used for question retrieval (Jeon et al., 2005; Xue et al., 2008; Lee et al., 2008; Bernhard and Gurevych, 2009). These word-based scores of bi-phrases, though not as effective in contextual selection, are more robust to noise and sparsity.

A sample of the resulting phrase translation examples is shown in Table 2, where the top 5 target phrases are translated from the source phrases according to the phrase-based translation model. For example, the term "explorer" used alone, most likely refers to a person who engages in scientific exploration, while the phrase "internet explorer" has a very different meaning.

3.4 Ranking Candidate Historical Questions

Unlike the word-based translation models, the phrase-based translation model cannot be interpolated with a unigram language model. Following (Sun et al., 2010; Gao et al., 2010), we resort to a linear ranking framework for question retrieval in which different models are incorporated as features.

We consider learning a relevance function of the following general, linear form:

$$Score(\mathbf{q}, D) = \boldsymbol{\theta}^T \cdot \boldsymbol{\Phi}(\mathbf{q}, D) \tag{19}$$

where the feature vector $\boldsymbol{\Phi}(\mathbf{q}, D)$ is an arbitrary function that maps $(\mathbf{q}, D)$ to a real value, i.e., $\boldsymbol{\Phi}(\mathbf{q}, D) \in \mathbb{R}$. $\boldsymbol{\theta}$ is the corresponding weight vector, we optimize this parameter for our evaluation metrics directly using the Powell Search algorithm (Paul et al., 1992) via cross-validation.

The features used in this paper are as follows:

- **Phrase translation features (PT)**: $\boldsymbol{\Phi}_{PT}(\mathbf{q}, D, A) = \log P(\mathbf{q}|D)$, where $P(\mathbf{q}|D)$ is computed using equations (12) to (15), and the phrase translation probability $P(\mathbf{w}|\mathbf{t})$ is estimated using equation (17).

- **Inverted Phrase translation features (IPT)**: $\boldsymbol{\Phi}_{IPT}(D, \mathbf{q}, A) = \log P(D|\mathbf{q})$, where $P(D|\mathbf{q})$ is computed using equations (12) to (15) except that we set $\mu_2 = 0$ in equation (15), and the phrase translation probability $P(\mathbf{w}|\mathbf{t})$ is estimated using equation (17).

- **Lexical weight feature (LW)**: $\boldsymbol{\Phi}_{LW}(\mathbf{q}, D, A) = \log P(\mathbf{q}|D)$, here $P(\mathbf{q}|D)$ is computed by equations (12) to (15), and the phrase translation probability is computed as lexical weight according to equation (18).

- **Inverted Lexical weight feature (ILW)**: $\boldsymbol{\Phi}_{ILW}(D, \mathbf{q}, A) = \log P(D|\mathbf{q})$, here $P(D|\mathbf{q})$ is computed by equations (12) to (15) except that we set $\mu_2 = 0$ in equation (15), and the phrase translation probability is computed as lexical weight according to equation (18).

- **Phrase alignment features (PA)**: $\boldsymbol{\Phi}_{PA}(\mathbf{q}, D, B) = \sum_2^K |a_k - b_{k-1} - 1|$, where B is a set of K bi-phrases, a_k is the start position of the phrase in D that was translated

into the kth phrase in queried question, and b_{k-1} is the end position of the phrase in D that was translated into the $(k-1)$th phrase in queried question. The feature, inspired by the distortion model in SMT (Koehn et al., 2003), models the degree to which the queried phrases are reordered. For all possible B, we only compute the feature value according to the Viterbi alignment, $\hat{B} = \arg\max_B P(\mathbf{q}, B|D)$. We find $\hat{B}$ using the Viterbi algorithm, which is almost identical to the dynamic programming recursion of equations (12) to (14), except that the *sum* operator in equation (13) is replaced with the *max* operator.

- **Unaligned word penalty features (UWP)**: $\mathbf{\Phi}_{UWP}(\mathbf{q}, D)$, which is defined as the ratio between the number of unaligned words and the total number of words in queried questions.

- **Language model features (LM)**: $\mathbf{\Phi}_{LM}(\mathbf{q}, D, A) = \log P_{LM}(\mathbf{q}|D)$, where $P_{LM}(\mathbf{q}|D)$ is the unigram language model with Jelinek-Mercer smoothing defined by equations (1) and (2).

- **Word translation features (WT)**: $\mathbf{\Phi}_{WT}(\mathbf{q}, D) = \log P(\mathbf{q}|D)$, where $P(\mathbf{q}|D)$ is the word-based translation model defined by equations (3) and (4).

4 Experiments

4.1 Data Set and Evaluation Metrics

We collect the questions from Yahoo! Answers and use the *getByCategory* function provided in Yahoo! Answers API[5] to obtain Q&A threads from the Yahoo! site. More specifically, we utilize the *resolved* questions under the top-level category at Yahoo! Answers, namely "Computers & Internet". The resulting question repository that we use for question retrieval contains 518,492 questions. To learn the translation probabilities, we use about one million question-answer pairs from another data set.[6]

In order to create the test set, we randomly select 300 questions for this category, denoted as "CI_TST". To obtain the ground-truth of question retrieval, we employ the Vector Space Model (VSM) (Salton et al., 1975) to retrieve the top 20 results and obtain manual judgements. The top 20 results don't include the queried question itself. Given a returned result by VSM, an annotator is asked to label it with "relevant" or "irrelevant". If a returned result is considered semantically equivalent to the queried question, the annotator will label it as "relevant"; otherwise, the annotator will label it as "irrelevant". Two annotators are involved in the annotation process. If a conflict happens, a third person will make judgement for the final result. In the process of manually judging questions, the annotators are presented only the questions. Table 3 provides the statistics on the final test set.

	#queries	#returned	#relevant
CI_TST	300	6,000	798

Table 3: Statistics on the Test Data

We evaluate the performance of our approach using **Mean Average Precision (MAP)**. We perform a significant test, i.e., a t-test with a default significant level of 0.05. Following the literature, we set the parameters $\lambda = 0.2$ (Cao et al., 2010) in equations (1), (3) and (5), and $\alpha = 0.8$ (Xue et al., 2008) in equation (6).

4.2 Question Retrieval Results

We randomly divide the test questions into five subsets and conduct 5-fold cross-validation experiments. In each trial, we tune the parameters μ_1 and μ_2 with four of the five subsets and then apply it to one remaining subset. The experiments reported below are those averaged over the five trials.

Table 4 presents the main retrieval performance. Row 1 to row 3 are baseline systems, all these methods use word-based translation models and obtain the state-of-the-art performance in previous work (Jeon et al., 2005; Xue et al., 2008). Row 3 is similar to row 2, the only difference is that TransLM only considers the question part, while Xue et al. (2008) incorporates the question part and answer part. Row 4 and row 5 are our proposed phrase-based translation model with maximum phrase length of five. Row 4 is phrase-based translation model purely based on question part, this model is equivalent to

[5] http://developer.yahoo.com/answers

[6] The Yahoo! Webscope dataset Yahoo answers comprehensive questions and answers version 1.0.2, available at http://reseach.yahoo.com/Academic_Relations.

#	Methods	Trans Prob	MAP
1	Jeon et al. (2005)	P_{pool}	0.289
2	TransLM	P_{pool}	0.324
3	Xue et al. (2008)	P_{pool}	0.352
4	P-Trans ($\mu_1 = 1, l = 5$)	P_{pool}	0.366
5	**P-Trans** ($l = 5$)	P_{pool}	**0.391**

Table 4: Comparison with different methods for question retrieval.

setting $\mu_1 = 1$ in equation (15). Row 5 is the phrase-based combination model which linearly combines the question part and answer part. As expected, different parts can play different roles: a phrase to be translated in queried questions may be translated from the question part or answer part. All these methods use pooling strategy to estimate the translation probabilities. There are some clear trends in the result of Table 4:

(1) Word-based translation language model (TransLM) significantly outperforms word-based translation model of Jeon et al. (2005) (row 1 vs. row 2). Similar observations have been made by Xue et al. (2008).

(2) Incorporating the answer part into the models, either word-based or phrase-based, can significantly improve the performance of question retrieval (row 2 vs. row 3; row 4 vs. row 5).

(3) Our proposed phrase-based translation model (P-Trans) significantly outperforms the state-of-the-art word-based translation models (row 2 vs. row 4 and row 3 vs. row 5, all these comparisons are statistically significant at $p < 0.05$).

4.3 Impact of Phrase Length

Our proposed phrase-based translation model, due to its capability of capturing contextual information, is more effective than the state-of-the-art word-based translation models. It is important to investigate the impact of the phrase length on the final retrieval performance. Table 5 shows the results, it is seen that using the longer phrases up to the maximum length of five can consistently improve the retrieval performance. However, using much longer phrases in the phrase-based translation model does not seem to produce significantly better performance (row 8 and row 9 vs. row 10 are not statistically significant).

#	Systems	MAP
6	P-Trans ($l = 1$)	0.352
7	P-Trans ($l = 2$)	0.373
8	P-Trans ($l = 3$)	0.386
9	P-Trans ($l = 4$)	0.390
10	P-Trans ($l = 5$)	0.391

Table 5: The impact of the phrase length on retrieval performance.

Model	#	Methods	Average	MAP
P-Trans ($l = 5$)	11	Initial	69	0.380
	12	TextRank	24	0.391

Table 6: Effectiveness of parallel corpus preprocessing.

4.4 Effectiveness of Parallel Corpus Preprocessing

Question-answer pairs collected from Yahoo! answers are very noisy, it is possible for translation models to contain "unnecessary" translations. In this paper, we attempt to identify and decrease the proportion of unnecessary translations in a translation model by using TextRank algorithm. This kind of "unnecessary" translation between words will eventually affect the bi-phrase translation.

Table 6 shows the effectiveness of parallel corpus preprocessing. Row 11 reports the average number of translations per word and the question retrieval performance when only stopwords [7] are removed. When using the TextRank algorithm for parallel corpus preprocessing, the average number of translations per word is reduced from 69 to 24, but the performance of question retrieval is significantly improved (row 11 vs. row 12). Similar results have been made by Lee et al. (2008).

4.5 Impact of Pooling Strategy

The correspondence of words or phrases in the question-answer pair is not as strong as in the bilingual sentence pair, thus noise will be inevitably introduced for both $P(\bar{\mathbf{a}}|\bar{\mathbf{q}})$ and $P(\bar{\mathbf{q}}|\bar{\mathbf{a}})$.

To see how much the pooling strategy benefit the question retrieval, we introduce two baseline methods for comparison. The first method (denoted as $P(\bar{\mathbf{a}}|\bar{\mathbf{q}})$) is used to denote the translation probability with the question as the source and the answer as

[7] http://truereader.com/manuals/onix/stopwords1.html

Model	#	Trans Prob	MAP
P-Trans ($l = 5$)	13	$P(\bar{\mathbf{a}}\|\bar{\mathbf{q}})$	0.387
	14	$P(\bar{\mathbf{q}}\|\bar{\mathbf{a}})$	0.381
	15	P_{pool}	0.391

Table 7: The impact of pooling strategy for question retrieval.

the target. The second (denoted as $P(\bar{\mathbf{a}}|\bar{\mathbf{q}})$) is used to denote the translation probability with the answer as the source and the question as the target. Table 7 provides the comparison. From this Table, we see that the pooling strategy significantly outperforms the two baseline methods for question retrieval (row 13 and row 14 vs. row 15).

5 Conclusions and Future Work

In this paper, we propose a novel phrase-based translation model for question retrieval. Compared to the traditional word-based translation models, the proposed approach is more effective in that it can capture contextual information instead of translating single words in isolation. Experiments conducted on real Q&A data demonstrate that the phrase-based translation model significantly outperforms the state-of-the-art word-based translation models.

There are some ways in which this research could be continued. First, question structure should be considered, so it is necessary to combine the proposed approach with other question retrieval methods (e.g., (Duan et al., 2008; Wang et al., 2009; Bunescu and Huang, 2010)) to further improve the performance. Second, we will try to investigate the use of the proposed approach for other kinds of data set, such as categorized questions from forum sites and FAQ sites.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (No. 60875041 and No. 61070106). We thank the anonymous reviewers for their insightful comments. We also thank Maoxi Li and Jiajun Zhang for suggestion to use the alignment toolkits.

References

A. Berger and R. Caruana and D. Cohn and D. Freitag and V. Mittal. 2000. Bridging the lexical chasm: statistical approach to answer-finding. In *Proceedings of SIGIR*, pages 192-199.

A. Berger and J. Lafferty. 1999. Information retrieval as statistical translation. In *Proceedings of SIGIR*, pages 222-229.

D. Bernhard and I. Gurevych. 2009. Combining lexical semantic resources with question & answer archives for translation-based answer finding. In *Proceedings of ACL*, pages 728-736.

P. F. Brown and V. J. D. Pietra and S. A. D. Pietra and R. L. Mercer. 1993. The mathematics of statistical machine translation: parameter estimation. *Computational Linguistics*, 19(2):263-311.

R. Bunescu and Y. Huang. 2010. Learning the relative usefulness of questions in community QA. In *Proceedings of EMNLP*, pages 97-107.

X. Cao and G. Cong and B. Cui and C. S. Jensen. 2010. A generalized framework of exploring category information for question retrieval in community question answer archives. In *Proceedings of WWW*.

D. Chiang. 2005. A hierarchical phrase-based model for statistical machine translation. In *Proceedings of ACL*.

H. Duan and Y. Cao and C. Y. Lin and Y. Yu. 2008. Searching questions by identifying questions topics and question focus. In *Proceedings of ACL*, pages 156-164.

J. Gao and X. He and J. Nie. 2010. Clickthrough-based translation models for web search: from word models to phrase models. In *Proceedings of CIKM*.

J. Jeon and W. Bruce Croft and J. H. Lee. 2005. Finding similar questions in large question and answer archives. In *Proceedings of CIKM*, pages 84-90.

R. Mihalcea and P. Tarau. 2004. TextRank: Bringing order into text. In *Proceedings of EMNLP*, pages 404-411.

P. Koehn and F. Och and D. Marcu. 2003. Statistical phrase-based translation. In *Proceedings of NAACL*, pages 48-54.

J. -T. Lee and S. -B. Kim and Y. -I. Song and H. -C. Rim. 2008. Bridging lexical gaps between queries and questions on large online Q&A collections with compact translation models. In *Proceedings of EMNLP*, pages 410-418.

F. Och. 2002. Statistical mahcine translation: from single word models to alignment templates. Ph.D thesis, RWTH Aachen.

F. Och and H. Ney. 2004. The alignment template approach to statistical machine translation. *Computational Linguistics*, 30(4):417-449.

J. M. Ponte and W. B. Croft. 1998. A language modeling approach to information retrieval. In *Proceedings of SIGIR*.

W. H. Press and S. A. Teukolsky and W. T. Vetterling and B. P. Flannery. 1992. *Numerical Recipes In C*. Cambridge Univ. Press.

S. Robertson and S. Walker and S. Jones and M. Hancock-Beaulieu and M. Gatford. 1994. Okapi at trec-3. In *Proceedings of TREC*, pages 109-126.

G. Salton and A. Wong and C. S. Yang. 1975. A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613-620.

X. Sun and J. Gao and D. Micol and C. Quirk. 2010. Learning phrase-based spelling error models from clickthrough data. In *Proceedings of ACL*.

K. Wang and Z. Ming and T-S. Chua. 2009. A syntactic tree matching approach to finding similar questions in community-based qa services. In *Proceedings of SIGIR*, pages 187-194.

X. Xue and J. Jeon and W. B. Croft. 2008. Retrieval models for question and answer archives. In *Proceedings of SIGIR*, pages 475-482.

C. Zhai and J. Lafferty. 2001. A study of smooth methods for language models applied to ad hoc information retrieval. In *Proceedings of SIGIR*, pages 334-342.