Extracting Question-Context-Answer Triples from Online Forums

Shilin Ding  Gao Cong  Chin-Yew Lin  Xiaoyan Zhu

Abstract

In this paper, we propose a new approach to extracting question-context-answer triples from online discussion forums. More specifically, we propose a general framework based on Conditional Random Fields (CRFs) for context and answer detection, and also extend the basic framework to utilize contexts for answer detection and to better accommodate the features of forums. Experimental results show that our techniques are very promising.

1 Introduction

Forums are web virtual spaces where people can ask questions, answer questions and participate in discussions. The availability of affluent thread discussions in forums has promoted increasing interests in knowledge acquisition and summarization for forum threads. Forum thread usually consists of an initiating post and a number of reply posts. The initiating post usually contains several questions and the reply posts usually contain answers to the questions and perhaps new questions. Forum participants are not physically co-present, and thus reply may not happen immediately after questions are posted. The asynchronous nature and multi-participants make multiple questions and answers interweaved together, which makes it more difficult to summarize.

In this paper, we address the problem of detecting question-context-answer triples from forums. Figure 1 gives an example of a forum thread with questions, contexts and answers annotated. It contains three question sentences, S3, S5 and S6. Sentences S1 and S2 are contexts of question 1 (S3). Sentence S4 is the context of questions 2 and 3, but not 1. Sentence S8 is the answer to question 3. One example of question-context-answer triple is (S4-S5-S10). As shown in the example, a forum question usually requires contextual information to provide background or constraints. Moreover, it sometimes needs contextual information to provide explicit link to its answers. For example, S8 is an answer of question 1, but they cannot be linked with any common word. Instead, S8 shares word pet with S1, which is a context of question 1, and thus S8 could be linked with question 1 through S1. We call contextual information the context of a question in this paper.

A summary of forum threads in the form of question-context-answer can not only highlight the main content, but also provide a user-friendly organization of threads, which will make the access to forum information easier.

Another motivation of detecting question-context-answer triples in forum threads is that it could be used to enrich the knowledge base of community-based question and answering (CQA) services such as Live QnA and Yahoo! Answers, where context is comparable with the question description while question corresponds to the question title. For example, there were about 700,000
questions in the Yahoo! Answers travel category as of January 2008. We extracted about 3,000,000 travel related questions from six online travel forums. One would expect that a CQA service with large QA data will attract more users to the service.

However, it is challenging to summarize forum threads into question-context-answer triples. First, detecting contexts of a question is important and non-trivial. We found that 74% of questions in our corpus containing 2,041 questions from 591 forum threads about travel need context. However, relative position information is far from adequate to solve the problem. For example, in our corpus 37% of sentences preceding questions are contexts and they only represent 20% of all correct contexts. To effectively detect contexts, the dependency between sentences is important. For example in Figure 1, both S1 and S2 are contexts of question 1. S1 could be labeled as context based on word similarity, but it is not easy to link S2 with the question directly. S1 and S2 are linked by the common word family, and thus S2 can be linked with question 1 through S1. The challenge here is how to model and utilize the dependency for context detection.

Second, it is difficult to link answers with questions. In forums, multiple questions and answers can be discussed in parallel and are interweaved together while the reply relationship between posts is usually unavailable. To detect answers, we need to handle two kinds of dependencies. One is the dependency relationship between contexts and answers, which should be leveraged especially when questions alone do not provide sufficient information to find answers; the other is the dependency between answer candidates (similar to sentence dependency described above). The challenge is how to model and utilize these two kinds of dependencies.

In this paper we propose a novel approach for summarizing forum threads into question-context-answer triples. We make the following contributions:

First, to our knowledge this is the first work on extracting question-context-answer triples from forum threads. We also found that context is very important for answer detection.

Second, we use classification method to identify questions from forum data as focuses of a thread, and then employ Linear Conditional Random Fields (CRFs) to identify contexts and answers, which can capture the relationships between contiguous sentences.

Third, to capture the dependency between contexts and answers, we introduce Skip-chain CRF model for answer detection. We also extend the basic model to 2D CRFs to model dependency between contiguous questions in a forum thread for context and answer identification.

Finally, we conducted experiments on forum data. Experimental results show that 1) our method for finding questions is effective; 2) Linear CRF model outperforms SVM and decision tree in both context and answer detection; 3) Skip-chain CRFs outperform Linear CRFs for answer finding, which demonstrates that context improves answer finding; 4) The combination of 2D CRFs and Skip-chain CRFs achieves better performance.

The rest of this paper is organized as follows: The next section discusses related work. Section 3 presents the proposed techniques. We evaluate our techniques in Section 4. Section 5 concludes this paper and discusses future work.

2 Related Work

There is some research on summarizing discussion threads and emails. Newman and Blitzer (2003) focused on clustering newsgroup messages into subtopic groups and extracting top-ranked sentences per group as summary. Motivated by the fact that one message may contain multiple issues, Zhou and Hovy (2005) segmented internet relay chat, clustered segments into subtopics, and identified responding segments of the first segment in each subtopic by assuming the first segment to be focus. Lam et al. (2002) employed a single document summarizer for email summarization. In (Nenkova and Bagga, 2003; Wan and McKeown, 2004; Rambow et al., 2004), email summaries were organized by extracting overview sentences as discussion issues. Carenini et al. (2007) leveraged both quotation relation and clue words for email summarization. In contrast, given a forum thread, we extract questions, their contexts, and their answers as summaries.

We also note the existing work on extracting knowledge from discussion threads. Huang et al. (2007) used SVM to extract input-reply pairs from forums for chatbot knowledge. Feng et al. (2006a)
used cosine similarity to match students’ query with reply posts for discussion-bot. Feng et al. (2006b) identified the most important message in online classroom discussion board. Our problem is quite different from the above work.

Detecting context for question in forums is related to the context detection problem raised in the QA roadmap paper commissioned by ARDA (Burger et al., 2006). To our knowledge, none of the previous work addresses the problem of context detection. The method of finding follow-up questions (Yang et al., 2006) from TREC context track could be adapted for context detection. However, the follow-up relationship is limited between questions while context is not.

Shrestha and McKeown (2004)’s work on email summarization is closer to our work. They used RIPPER as a classifier to detect interrogative questions and their answers and used the resulting question and answer pairs as summaries. However, it did not consider contexts of questions and dependency between answer sentences.

Extensive research has been done in question-answering, e.g. (Berger et al., 2000; Jeon et al., 2005; Cui et al., 2005; Harabagiu and Hickl, 2006; Dang et al., 2007). They mainly focus on constructing answer for certain types of question from a large document collection, and usually apply sophisticated linguistic analysis to both questions and the documents in the collection. In our scenario, we not only need to find answers for various types of questions in forum threads but also the questions themselves.

3 Question-context-answer Triple Detection

In this section we will first introduce the problem of finding question-context-answer triples from forums, and then present our approach.

Problem statement. A question is a linguistic expression used by a questioner to request information in the form of an answer. A question usually contains question focus, i.e. question concept that embodies information expectation of question (Lehnert, 1977), and constraints. The sentence containing question focus is called question anchor or simply question and the sentences containing only constraints are called context. Context provides constraint or background information to question.

As discussed in Introduction, identifying question-context-answer triples from forums is nontrivial. We approach the problem by first identifying questions in a thread, and then identifying the context and answer of every question within a uniform framework. In this section, we will first briefly present an approach to question detection, and then focus on context and answer detection.

3.1 Question Detection.

For question detection in forums, rules, such as question mark and 5W1H words, are not adequate. With question mark as an example, we find that 30% questions do not end with question marks while 9% sentences ending with question marks are not questions in a corpus. To complement the inadequacy of simple rules, in this paper we build a SVM classifier to detect questions using the features exploited in (Shrestha and McKeown, 2004).

3.2 Context and answer detection

Given a thread and a set of $m$ detected questions $\{Q_i\}_{i=1}^m$, our task is to find the contexts and answers for each question. We first discuss using linear CRFs model for context and answer detection, and then extend the basic framework to Skip-chain CRFs and 2D CRFs to better model our problem. Finally, we will briefly introduce CRF models and the features that we used for CRF model.

3.2.1 Using linear CRFs

For ease of presentation, we discuss detecting contexts of the questions using linear CRF model. The model could be easily extended to answer detection (by using different features).

Context detection. As discussed in Introduction that context detection cannot be trivially solved by position information (See Section 4.2 for details), and dependency between sentences is important for context detection. Recall that in Figure 1 S2 could be labeled as context of Q1 if we consider the dependency between S2 and S1, and that between S1 and Q1, while it is difficult to establish connection between S2 and Q1 without S1. Table 1 shows that the correlation between the labels of contiguous sentences is significant. In other words, when a sentence $Y_l$’s previous $Y_{l-1}$ is not a context ($Y_{l-1} \neq C$)
edges can capture the dependency between two contiguous nodes. The observation sequence $x = <x_1, x_2, \ldots, x_t>$, where $t$ is the number of sentences in a thread, represents predictors (to be described in Section 3.2.5), and the tag sequence $y = <y_1, \ldots, y_t>$, where $y_i \in \{C, P\}$, determines whether a sentence is plain text $P$ or context $C$ of question $Q_i$.

**Answer detection.** Answers usually appear in the posts after the post containing the question. We observed that paragraph is usually a good segment for answer while the proposed approach is applicable to other kinds of segments. There are also strong dependencies between contiguous answer segments. Thus, position information and similarity method are not adequate for answer detection. To cope with the dependency between contiguous answer segments, we employ linear CRF models for answer detection.

### 3.2.2 Leveraging context for answer detection using Skip-chain CRFs

We observed 74% questions lack contextual information in our corpus. As discussed in introduction, the constraints or background information provided by context are very useful to link question and answers. Therefore, contexts should be leveraged to detect answers. The linear CRF model can capture the dependency between contiguous sentences. However, it cannot capture the long distance dependency between contexts and answers.

One straightforward method of leveraging context is to detect contexts and answers in two phases, i.e. to first identify contexts, and then label answers using both the context and question information (e.g. the similarity between context and answer can be used as features in CRFs). The two-phase procedure, however, still cannot capture the non-local dependency between contexts and answers in a thread.

To model the long distance dependency between contexts and answers, we will use Skip-chain CRF model to detect context and answer together. Skip-chain CRF model is applied for entity extraction and meeting summarization (Sutton and McCallum, 2006; Galley, 2006). The graphical representation of a Skip-chain CRF given in Figure 2(b) consists of two types of edges: linear-chain ($y_t$ to $y_{t-1}$) and skip-chain edges ($y_t$ to $y_n$).

<table>
<thead>
<tr>
<th>Contiguous sentences</th>
<th>$y_t = C$</th>
<th>$y_t \neq C$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$y_{t-1} = C$</td>
<td>1,191</td>
<td>1,366</td>
</tr>
<tr>
<td>$y_{t-1} \neq C$</td>
<td>1,377</td>
<td>62,446</td>
</tr>
</tbody>
</table>

Table 1: Contingency table ($\chi^2 = 13.044$, p-value < 0.001)
connection between candidate pairs with high probability of being context and answer of a question. To introduce skip-chain edges between any pairs of non-contiguous sentences will be computationally expensive for Skip-chain CRFs, and also introduce noise. To make the cardinality and number of cliques in the graph manageable and also eliminate noisy edges, we would like to generate edges only for sentence pairs with high possibility of being context and answer. We explain how do we achieve this next. Given a question \( Q_i \) in post \( P_j \) of a thread with \( n \) posts, its contexts usually occur within post \( P_j \) or before \( P_j \) while answers appear in the posts after \( P_j \). In this paper, we will establish an edge between each candidate answer \( v \) and one candidate context in \( \{ P_k \}_{k=1}^n \) such that they have the highest possibility of being a context-answer pair of question \( Q_i \). We use the product of \( \text{sim}(x_u, Q_i) \) and \( \text{sim}(x_v, \{ x_u, Q_i \}) \) to estimate the possibility of being a context-answer pair for \( (u, v) \).

\[
\arg\max_{u \in \{ P_k \}_{k=1}^n} \text{sim}(x_u, Q_i) \cdot \text{sim}(x_v, \{ x_u, Q_i \}) \tag{1}
\]

Table 2 shows that \( y_u \) and \( y_v \) in the skip chain generated by our heuristics influence each other.

Skip-chain CRFs improve the performance of answer detection due to the introduced skip-chain edges that represent the joint probability conditioned on the question, which is exploited by skip-chain feature function: \( f(y_u, y_v, Q_i, x) \).

### 3.2.3 Using 2D CRF Model

Both Linear CRFs and Skip-chain CRFs label the contexts and answers for each question in separate passes by assuming that questions in a thread are independent. Actually the assumption does not hold in many cases. Let us look at an example. As in Figure 1, Sentence S10 is an answer for both question 2 and question 3. S10 could be recognized as the answer of question 2 due to the shared word traffic, but there is no direct relation between question 3 and S10. To label S10, we need consider the dependency relation between question 2 and 3. In other words, the question-answer relation between question 3 and S10 can be captured by a joint modeling of the dependency among S10, question 2 and question 3. The labels of the same sentence for two contiguous questions in a thread would be conditioned on the dependency relationship between the questions. Such a dependency cannot be captured by both Linear CRFs and Skip-chain CRFs.

To capture the dependency between the contiguous questions, we employ 2D CRFs to help context and answer detection. 2D CRF model is used in (Zhu et al., 2005) to model the neighborhood dependency in blocks within a web page. As shown in Figure 2(c), 2D CRF models the labeling task for all questions in a thread. The \( i \)-th row in a grid corresponds to one pass of Linear CRF model (or Skip-chain model) which labels contexts and answers for question \( Q_i \). The vertical edges in the figure represent the joint probability conditioned on the contiguous questions, which will be exploited by 2D feature function: \( f(y_i, y_{i+1}, Q_i, Q_{i+1}, x) \). Thus, the information generated in single CRF chain could be propagated over the whole grid. In this way, context and answer detection for all questions in the thread could be modeled together.

### 3.2.4 Conditional Random Fields (CRFs)

The Linear, Skip-Chain and 2D CRFs can be generalized as pairwise CRFs, which have two kinds of cliques in graph \( G \): 1) node \( y_t \) and 2) edge \( (y_u, y_v) \). The joint probability is defined as:
\[ p(y|x) = \frac{1}{Z(x)} \exp \left\{ \sum_{k,t} \lambda_k f_k(y_t, x) + \sum_{k,t} \mu_k g_k(y_u, y_v, x) \right\}, \]

where \( Z(x) \) is the normalization factor, \( f_k \) is the feature on nodes, \( g_k \) is on edges between \( u \) and \( v \), and \( \lambda_k \) and \( \mu_k \) are parameters.

Linear CRFs are based on the first order Markov assumption that the contiguous nodes are dependent. The pairwise edges in Skip-chain CRFs represent the long distance dependency between the skipped nodes, while the ones in 2D CRFs represent the dependency between the horizontal nodes.

**Inference and Parameter Estimation.** For linear CRFs, dynamic programming is used to compute the maximum a posteriori (MAP) of \( y \) given \( x \). However, for more complicated graphs with cycles, exact inference needs the junction tree representation of the original graph and the algorithm is exponential to the treewidth. For fast inference, loopy Belief Propagation (Pearl, 1988) is implemented.

Given the training Data \( D = \{x^{(i)}, y^{(i)}\}_{i=1}^n \), the parameter estimation is to determine the parameters based on maximizing the log-likelihood \( L_\lambda = \sum_{i=1}^n \log p(y^{(i)}|x^{(i)}) \). In linear CRF model, dynamic programming and L-BFGS \(^1\) can be used to optimize objective function \( L_\lambda \), while for complicated CRFs, Loopy BP are used instead to calculate the marginal probability.

### 3.2.5 Features used in CRF models

The main features used in linear CRF models for context detection are listed in Table 3. We next briefly introduce them.

The similarity feature is to capture the words similarity and semantic similarity between candidate contexts and answers. The semantic similarity is computed based on Wu and Palmer’s measure (Wu and Palmer, 1994) using WordNet (Fellbaum, 1998). The similarity between contiguous sentences will be used to capture the dependency for CRFs. In addition, to bridge the lexical gaps between question and context, we learned top-3 context terms for each question term from 300,000 question-description pairs obtained from Yahoo! Answers using mutual

---

1. L-BFGS stands for limited memory Broyden-Fletcher-Goldfarb-Shanno.

---

Table 3: Features for linear CRFs. Unless otherwise mentioned, we refer to features of the sentence whose label to be predicted

<table>
<thead>
<tr>
<th>Similarity features:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cosine similarity with the question</td>
</tr>
<tr>
<td>Similarity with the question using WordNet</td>
</tr>
<tr>
<td>Cosine similarity between contiguous sentences</td>
</tr>
<tr>
<td>Similarity between contiguous sentences using WordNet</td>
</tr>
<tr>
<td>Cosine similarity with the expanded question using the lexical matching words</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Structural features:</th>
</tr>
</thead>
<tbody>
<tr>
<td>The relative position to current question</td>
</tr>
<tr>
<td>Is the author the same with that of the question.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Discourse and lexical features:</th>
</tr>
</thead>
<tbody>
<tr>
<td>The number of Pronouns in the question</td>
</tr>
<tr>
<td>The presence of fillers, fluency devices (e.g. “uh”, “ok”)</td>
</tr>
<tr>
<td>The presence of acknowledgment tokens</td>
</tr>
<tr>
<td>The number of non-stopswords</td>
</tr>
<tr>
<td>Whether the question has a noun or not</td>
</tr>
<tr>
<td>Whether the question has a verb or not</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Discourse and lexical features:</th>
</tr>
</thead>
<tbody>
<tr>
<td>The number of Pronouns in the question</td>
</tr>
<tr>
<td>The presence of fillers, fluency devices (e.g. “uh”, “ok”)</td>
</tr>
<tr>
<td>The presence of acknowledgment tokens</td>
</tr>
<tr>
<td>The number of non-stopswords</td>
</tr>
<tr>
<td>Whether the question has a noun or not</td>
</tr>
<tr>
<td>Whether the question has a verb or not</td>
</tr>
</tbody>
</table>

4 Experiments

### 4.1 Experimental setup

**Corpus.** We obtained about 1 million threads from TripAdvisor \(^2\) forum and randomly selected 591 fo-
Table 4: Performance of Question Detection

<table>
<thead>
<tr>
<th>Feature</th>
<th>Prec(%)</th>
<th>Rec(%)</th>
<th>F₁(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>5W-1H words</td>
<td>69.98</td>
<td>14.95</td>
<td>24.63</td>
</tr>
<tr>
<td>Question Mark</td>
<td>91.25</td>
<td>69.85</td>
<td>79.12</td>
</tr>
<tr>
<td>RIPPER</td>
<td>88.84</td>
<td>75.81</td>
<td>81.76</td>
</tr>
<tr>
<td>Our</td>
<td>88.75</td>
<td>87.03</td>
<td>87.85</td>
</tr>
</tbody>
</table>

Table 5: Context and Answer Detection

<table>
<thead>
<tr>
<th>Model</th>
<th>Prec(%)</th>
<th>Rec(%)</th>
<th>F₁(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Context Detection</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SVM</td>
<td>61.76</td>
<td>58.89</td>
<td>60.27</td>
</tr>
<tr>
<td>C4.5</td>
<td>60.09</td>
<td>54.13</td>
<td>56.95</td>
</tr>
<tr>
<td>Linear CRF</td>
<td>63.25</td>
<td>69.17</td>
<td>66.07</td>
</tr>
<tr>
<td>Answer Detection</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SVM</td>
<td>61.36</td>
<td>46.81</td>
<td>53.31</td>
</tr>
<tr>
<td>C4.5</td>
<td>68.36</td>
<td>40.55</td>
<td>50.90</td>
</tr>
<tr>
<td>Linear CRF</td>
<td>78.85</td>
<td>49.37</td>
<td>59.76</td>
</tr>
</tbody>
</table>

Table 6: Using position information for detection

<table>
<thead>
<tr>
<th>position</th>
<th>Prec(%)</th>
<th>Rec(%)</th>
<th>F₁(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Context Detection</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Previous One</td>
<td>37.33</td>
<td>20.48</td>
<td>26.45</td>
</tr>
<tr>
<td>Previous All</td>
<td>33.66</td>
<td>79.40</td>
<td>47.28</td>
</tr>
<tr>
<td>Answer Detection</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Following One</td>
<td>59.34</td>
<td>21.44</td>
<td>31.50</td>
</tr>
<tr>
<td>Following All</td>
<td>26.71</td>
<td>100</td>
<td>42.12</td>
</tr>
</tbody>
</table>

4.2 Experimental results

Question Detection. The experiment is to evaluate the performance of our question detection method against simple rules and the method used in (Shrestha and McKeown, 2004), denoted as RIPPER. The results are given in Table 5. The first two rows show the results of simple rules. The rule 5W-1H words is that a sentence is a question if it begins with 5W-1H words; The rule Question Mark is that a sentence is a question if it ends with question mark. Although Question Mark achieves the best precision, its recall is low. Our method outperforms the simple rules in terms of F₁-score. Our method differs from RIPPER in that we adopt SVM model while RIPPER was used in (Shrestha and McKeown, 2004).

Linear CRFs for Context and Answer Detection. This experiment is to evaluate Linear CRF model (Section 3.2.1) for context and answer detection by comparing with SVM and C4.5(Quinlan, 1993). For SVM, we use SVMlight(Joachims, 1999) and report the best SVM result when using linear or polynomial kernels. For context detection, SVM and C4.5 use the same set of features. For answer detection, for SVM and C4.5 we add the similarity between real context and answer as extra features; otherwise, they failed. As shown in Table 5, Linear CRF model outperforms SVM and C4.5 for both context and answer detection, even if Linear CRF did not use any context information for answer finding. The main reason for the improvement is that CRF models can capture the sequential dependency between segments in forums as discussed in Section 3.2.1.

We next report a baseline of context detection using previous sentences in the same post with its question since contexts often occur in the question post or preceding posts. Similarly, we report a baseline of answer detecting using following segments of a question as answers. The results given in Table 6 show that location information is far from adequate to detect contexts and answers.

The usefulness of contexts. This experiment is to evaluate the usefulness of contexts in answer detection, by adding the similarity between the context (obtained with different methods) and candidate answer as an extra feature for CRFs. Table 7 shows the impact of context on answer detection using Linear CRFs. L-CRF+context uses the context found by Linear CRFs, and performs better than Linear CRF without context. We also found that the performance
Table 7: Contextual Information for Answer Detection

<table>
<thead>
<tr>
<th>Model</th>
<th>Prec(%)</th>
<th>Rec(%)</th>
<th>F₁(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>No context</td>
<td>78.85</td>
<td>49.37</td>
<td>59.76</td>
</tr>
<tr>
<td>L-CRF+context</td>
<td>79.56</td>
<td>55.81</td>
<td>64.64</td>
</tr>
<tr>
<td>Prev. sentence</td>
<td>79.48</td>
<td>51.73</td>
<td>61.71</td>
</tr>
<tr>
<td>Real context</td>
<td>80.09</td>
<td>57.36</td>
<td>65.88</td>
</tr>
</tbody>
</table>

Table 8: Skip-chain and 2D CRFs for answer detection

<table>
<thead>
<tr>
<th>Model</th>
<th>Prec(%)</th>
<th>Rec(%)</th>
<th>F₁(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>L-CRF+context</td>
<td>79.56</td>
<td>55.81</td>
<td>64.64</td>
</tr>
<tr>
<td>Skip-chain</td>
<td>78.40</td>
<td>67.74</td>
<td>72.38</td>
</tr>
<tr>
<td>2D</td>
<td>76.95</td>
<td>65.66</td>
<td>70.61</td>
</tr>
<tr>
<td>2D+Skip-chain</td>
<td>81.81</td>
<td>69.76</td>
<td>75.31</td>
</tr>
</tbody>
</table>

of L-CRF+context is close to that using real context, while it is better than CRFs using the previous sentence as context. The results clearly shows that contextual information greatly improves the performance of answer detection. This was also observed for other classification methods in our experiments: SVM and C4.5 (in Table 5) failed if we did not use context.

**Improved Models.** This experiment is to evaluate the effectiveness of Skip-Chain CRFs (Section 3.2.2) and 2D CRFs (Section 3.2.3) for our tasks. The results are given in Table 8. As expected, Skip-chain CRFs outperform L-CRF+context since Skip-chain CRFs can model the inter-dependency between contexts and answers while in L-CRF+context the context can only be reflected by the features on the observations. We also observed that 2D CRFs improves the performance of L-CRF+context and we achieved the best performance if we combine the 2D CRFs and Skip-chain CRFs. For context detection, there is slightly improvement, e.g. Precision (64.48%) Recall (71.51%) and F₁-score (67.79%).

**Evaluating Features.** We also evaluated the contributions of each category of features in Table 3 to context detection. We found that similarity features are the most important and structural feature the next. We also observed the same trend for answer detection. We omit the details here due to space limitation.

As a summary, 1) our question detection method is indeed effective; 2) our CRF model outperforms SVM and C4.5 for both context and answer detections; 3) context is very useful in answer detection; 4) the Skip-chain CRF method is effective in leveraging context for answer detection; and 5) the combination of 2D CRFs and Skip-chain CRF achieves the best performance.

5 Discussions and Conclusions

We presented a new approach to detecting question-context-answer triples in forums with good performance. We next discuss our experience not covered by the experiments, and future work.

We found our methods often cannot identify questions expressed by imperative sentences in question detection task, e.g. “recommend a restaurant in New York”. This would call for future work. We also observed that factoid questions, one of focuses in the TREC QA community, take less than 10% question in our corpus. It would be interesting to revisit QA techniques to process forum data.

Since contexts of questions are largely unexplored in previous work, we analyze the contexts in our corpus and classify them into three categories: 1) context contains the main content of question while question contains no constraint, e.g. “i will visit NY at Oct, looking for a cheap hotel but convenient. Any good suggestion?”; 2) contexts explain or clarify part of the question, such as a definite noun phrase, e.g. ‘We are going on the Taste of Paris. Does anyone know if it is advisable to take a suitcase with us on the tour., where the first sentence is to describe the tour; and 3) contexts provide constraint or background for question that is syntactically complete, e.g. “We are interested in visiting the Great Wall(and flying from London). Can anyone recommend a tour operator.” In our corpus, about 26% questions do not need context, 12% questions need Type 1 context, 32% need Type 2 context and 30% Type 3. We found that our techniques often do not perform well on Type 3 questions.

Other future work includes: 1) to summarize multiple threads using the triples extracted from individual threads. This could be done by clustering question-context-answer triples; 2) to use the traditional text summarization techniques to summarize the multiple answer segments; 3) to integrate the Question Answering techniques as features of our framework to further improve answer finding; and 4) to reformulate questions using its context to generate more user-friendly questions for CQA services.
References

A. Berger, R. Caruana, D. Cohn, D. Freitag, and V. Mit- 
tal. 2000. Bridging the lexical chasm: statistical ap-
proaches to answer-finding. In Proceedings of SIGIR.
J. Burger, C. Cardie, V. Chaudhri, R. Gaizauskas, 
S. Harabagiu, D. Israel, C. Jacquemin, C. Lin, 
S. Maiorano, G. Miller, D. Moldovan, B. Ogden, 
J. Prager, E. Riloff, A. Singhal, R. Shrihari, T. Strza-
lkowski16, E. Voorhees, and R. Weishedel. 2006. Is-
issues, tasks and program structures to roadmap research 
in question and answering (qna). ARAD: Advanced 
Research and Development Activity (US).
G. Carenini, R. Ng, and X. Zhou. 2007. Summarizing 
email conversations with clue words. In Proceedings 
of WWW.
H. Cui, R. Sun, K. Li, M. Kan, and T. Chua. 2005. Quest-
ion answering passage retrieval using dependency rela-
tions. In Proceedings of SIGIR.
H. Cunningham, D. Maynard, K. Bontcheva, and 
V. Tablan. 2002. Gate: A framework and graphical 
development environment for robust nlp tools and ap-
lications. In Proceedings of ACL.
H. Dang, J. Lin, and D. Kelly. 2007. Overview of the 
trec 2007 question answering track. In Proceedings of 
TREC.
C. Fellbaum, editor. 1998. WordNet: An Electronic Lex-
ical Database (Language, Speech, and Communication).
The MIT Press, May.
D. Feng, E. Shaw, J. Kim, and E. Hovy. 2006a. An intel-
ligent discussion-bot for answering student queries in 
threaded discussions. In Proceedings of IUI.
D. Feng, E. Shaw, J. Kim, and E. Hovy. 2006b. Learning 
to detect conversation focus of threaded discussions. 
In Proceedings of HLT-NAACL.
M. Galley. 2006. A skip-chain conditional random field 
for ranking meeting utterances by importance. In Pro-
ceedings of EMNLP.
S. Harabagiu and A. Hickl. 2006. Methods for using tex-
tual entailment in open-domain question answering. 
In Proceedings of ACL.
J. Huang, M. Zhou, and D. Yang. 2007. Extracting chat-
bot knowledge from online discussion forums. In Pro-
ceedings of IJCAI.
questions in large question and answer archives. In 
Proceedings of CIKM.
T. Joachims. 1999. Making large-scale support vector 
machine learning practical. MIT Press, Cambridge, 
MA, USA.
J. Lafferty, A. McCallum, and F. Pereira. 2001. Con-
ditional random fields: Probabilistic models for seg-
menting and labeling sequence data. In Proceedings of 
ICML.
Exploiting e-mail structure to improve summarization.
Ph.D. thesis, New Haven, CT, USA.
A. McCallum and W. Li. 2003. Early results for named 
entity recognition with conditional random fields, fea-
ture induction and web-enhanced lexicons. In Pro-
A. Nenkova and A. Bagga. 2003. Facilitating email 
thread access by extractive summary generation. In 
Proceedings of RANLP.
P. Newman and J. Blitzer. 2003. Summarizing archived 
discussions: A beginning. In Proceedings of IUI.
J. Pearl. 1988. Probabilistic reasoning in intelligent sys-
tems: networks of plausible inference. Morgan Kauf-
mann Publishers Inc., San Francisco, CA, USA.
J. Quinlan. 1993. C4.5: programs for machine learn-
ing. Morgan Kaufmann Publishers Inc., San Fran-
cisco, CA, USA.
HLT-NAACL.
F. Sha and F. Pereira. 2003. Shallow parsing with condi-
tional random fields. In HLT-NAACL.
L. Shrestha and K. McKeown. 2004. Detection of 
question-answer pairs in email conversations. In Pro-
ceedings of COLING.
C. Sutton and A. McCallum. 2006. An introduction to 
conditional random fields for relational learning. In 
Lise Getoor and Ben Taskar, editors, Introduction to 
summaries of ongoing email thread discussions. In 
Proceedings of COLING.
Z. Wu and M. S. Palmer. 1994. Verb semantics and lexi-
cal selection. In Proceedings of ACL.
F. Yang, J. Feng, and G. Fabbrianzio. 2006. A data 
driven approach to relevancy recognition for contex-
tual question answering. In Proceedings of the Inter-
active Question Answering Workshop at HLT-NAACL 
2006.
culture: The summarization of technical internet relay 
chats. In Proceedings of ACL.
conditional random fields for web information extrac-
tion. In Proceedings of ICML.