An Alignment-based Surface Pattern for a Question Answering System

Cheng-Lung Sung[1,2], Cheng-Wei Lee[2], Hsu-Chun Yen[1], Wen-Lian Hsu[2]
[1]Dept. of Electrical Engineering, National Taiwan University
[2]Institute of Information Science, Academia Sinica
{clsung,aska,hsu}@iis.sinica.edu.tw; yen@cc.ee.ntu.edu.tw

Abstract

In this paper, we propose an alignment-based surface pattern approach, called ABSP, which integrates semantic information into syntactic patterns for question answering (QA). ABSP uses surface patterns to extract important terms from questions, and constructs the terms' relations from sentences in the corpus. The relations are then used to filter appropriate answer candidates. Experiments show that ABSP can achieve high accuracy and can be incorporated into other QA systems that have high coverage. It can also be used in cross-lingual QA systems. The approach is both robust and portable to other domains.

1. Introduction

Interest in research on question answering (QA) systems, such as those created for the TREC (Text REtrieval Conference) QA task and NTCIR Cross Language Question Answering (NTCIR CLQA), has grown rapidly in recent year. Many open domain QA systems analyze an input question to determine the desired type of answer [4]. However, among the sixty-five participating systems in the TREC-10 Question Answering competition, the winning system used just one resource: a fairly extensive list of surface patterns [4, 10]. With the development of natural language processing, more QA researchers are employing semantic techniques to obtain more accurate answers.

Surface patterns are syntactic patterns that connect answers and question keywords. A number of online QA systems use patterns to deal with users' questions and extract answers. For example, Ravichandran and Hovy[4, 10] use surface patterns such as "<NAME> was born on <DATE>" and "<NAME>(<BIRTHDATE>-" to answer BIRTHDATE questions (When was X born?), while Ion provides three different linguistic patterns to extract relevant information[9]. Soubbotin [13] used richer patterns (including predefined string sequences, and unordered combination of strings and definition patterns) to answer questions and won the TREC-2001 competition.

However, since none of the above patterns include semantic information, they are called "poor-knowledge approaches" [11]. In other words, they cannot extract precise answers or relevant information without analyzing questions and answers semantically.

There has been some progress in adding semantic representations to linguistic patterns in order to improve the accuracy of answers. Maximiliano [11] proposed a type of semantic pattern that uses EuroWordNet as a lexical database. However, the pattern cannot provide semantic information itself and cannot represent the constraints on a specific part of a sentence. Steffen et al. [14] proposed a different kind of semantic pattern, which can be used for communications between semantic web developers as well as for mapping and reusing different target languages. However, it is not easy to use the pattern for question generation and representation. Furthermore, since it is designed primarily for professional use, it is difficult to use without domain knowledge.

Although surface patterns are simple and accurate, a number of issues need to be addressed. First, more finer-grained question types should be defined. For example, in addition to using a DATE question type, we may need more date-related question types, such as BIRTHDATE and PUBLICATIONDATE. However, this would increase the workload of the question classification process. Second, the method cannot deal with questions that have multiple keywords. Third, the method cannot handle cases where the information for validating the answer is spread over several passages. Fourth, since it requires exact matches, the method cannot be applied when there is a high language variation. Our proposed surface pattern method, ABSP, can resolve the first three problems. ABSPs are generated from question-answer pairs regardless of the question type. The surface patterns, which are automatically generated and selected from training data for any kind of question type, can capture important relations between a question's terms and the correct answer. In situations involving multiple question keywords and multiple passages, several ABSPs are used together to calculate a score for an answer. They can be
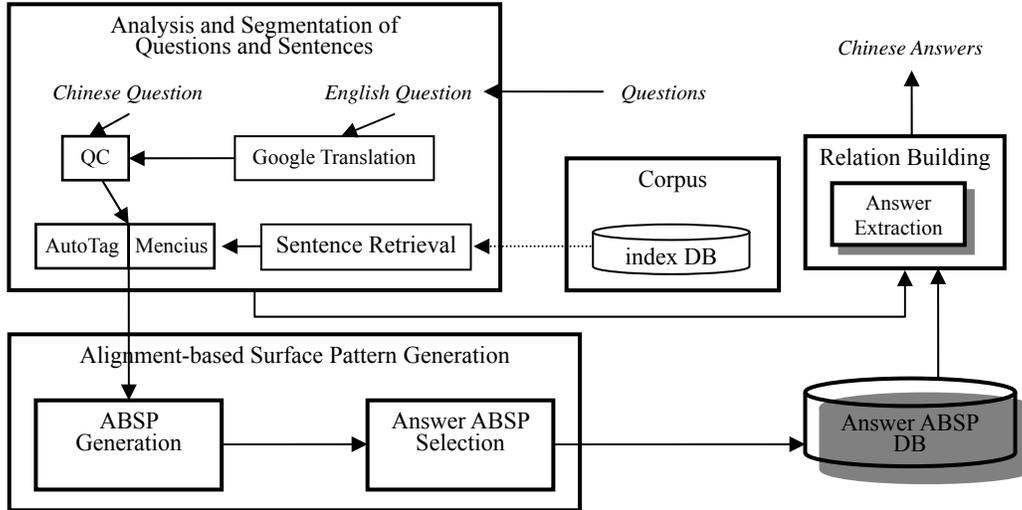
Figure 1. Workflow of proposed E-C, C-C QA system

used in cross-language QA systems. The proposed method is based on sequence alignment. It incorporates local alignment algorithms that have been thoroughly studied by Smith and Waterman [12], and uses dynamic programming to align sentence pairs. Then, we extract similar (syntactic and semantic) parts of the sentences as surface patterns.

The remainder of the paper is organized as follows: Section 2 describes the proposed ABSP approach, including semantic and syntactic pattern matching, and how we use the *ABSPs* for answer extraction. In Section 3, we discuss the experimental results. Finally, in Section 4, we summarize the present work and indicate the direction of our future research.

2. The workflow of using the ABSP Approach

To generate ABSPs, the system first retrieves and processes relevant sentences for all training questions from a corpus, and uses them to generate surface patterns.

The newly generated patterns, which can be filtered based on the question-answer pair as Answer ABSPs. The Answer ABSP is particularly useful for matching the keywords of new questions. Once the keywords of a question have been identified, the system matches them with existing Answer ABSPs, constructs their relations, and extracts the answers automatically.

The workflow chart of the question answering system is shown in Figure 1. It is comprised of four major processes: (1) Analysis and segmentation of questions and sentences, (2) The ABSP generation, (3) Answer ABSP Selection and (4) Relation Building and Answer extraction.

2.1. Analysis and Segmentation of Questions and Sentences

In our experiment, we apply the proposed approach to Chinese-Chinese QA and English-Chinese QA. To deal with English questions, we incorporate Google Translation [3] to translate English questions into Chinese. For Chinese sentences, as Chinese written texts do not contain word delimiters, we incorporate a Chinese segmentation tool, called AutoTag [1], to break a sentence into segments comprised of words and part-of-speech (POS) tags. We also use Mencius [15] to tag named entity (NE) words; the remaining words are tagged "O (others)". For example, the sentence "2000 年奧運在雪梨舉行 (2000 Olympics was held in Sydney)" would be segmented into "2000 年/Nd/TIME 奧運/Nb/O 在/P/O 雪梨/Nc/LOCATION 舉行/VC/O".

To select sentences for a given question, we use Lucene [8], an open source information retrieval engine. We filter out stop-words from question segments, and utilize the question segments to form Lucene queries. For each question, the top 200 sentences are chosen for answer extraction.

We adopt an integrated knowledge-based and machine learning approach for Chinese question classification. The approach use InfoMap [5] as the knowledge-based approach, which uses syntactic rules to model Chinese questions, and adopt SVM (Support Vector Machines) [16] as the machine learning approach for a large collection of labeled Chinese questions. Each question is classified into a question type or types by InfoMap and the SVM module. Then, the integrated module selects the question type with the highest confidence score. A detailed description of the question classification scheme can be found in [2].

Table 1. Example of an aligned sentence pair with the resulting ABSP.

榮獲/**VJ**/O	諾貝爾/**Nb**/ORG 和平獎/**Na**/O 的/DE/O 南韓/Nc/**LOC**		總統/**Na**/OCC	是/SHI/O 金大中/Nb/**PER**
參加/**VC**/O 2000 年/Nd/TIME 兩韓/**Nc**/LOC 高峰會/**Na**/O 的/DE/O 北韓/Nc/**LOC** 領導人/**Na**/O			是/SHI/O 金正日/Nb/**PER**	
V — **N** **Na** 的		**LOC**	**Na** 是	**PER**

2.2. ABSP Generation

Sequence alignment is the process that finds similar sequences in a pair of sentences. Pair-wise sequence alignment algorithms that generate templates and match them against new text have been researched extensively. Such methods are useful for extracting protein-protein interactions from biomedical texts annotated with parts-of-speech (POS)[6]. The sequences are padded with gaps so that similar characters can be aligned as closely as possible. Because we need surface patterns extracted from sentences that have certain morphological similarities, we employ local alignment techniques [12] to generate surface patterns.

To apply the local alignment algorithm, we first perform word segmentation and tagging, which we described in Section 2.1. The novelty of the proposed alignment algorithm is that it utilizes a similarity function. Our ABSPs contain named entity (NE) as semantic tag, and POS as syntactic tag. Consider two sequences $X = (x_1, x_2, \ldots, x_n)$ and $Y = (y_1, y_2, \ldots, y_m)$ defined over the alphabet that consists of four kinds of tag: NE tags, POS tags, a raw word tag for every single word, and a tag "-" for a gap. We assign a scoring function, F, to measure the similarity of X and Y. $F(i, j)$ is defined as the score of the optimal alignment between the initial segment from x_1 to x_i of X and the initial segment from y_1 to y_j of Y.

$F(i, j)$ is recursively calculated as follows:

$$F(i, 0) = 0, F(0, j) = 0, x_i, y_j \in \Sigma, \quad (i)$$

$$F(i, j) = \max \begin{cases} 0, \\ F(i-1, j-1) + s(x_i, y_j), \\ F(i-1, j) + s(x_i, '-') \\ F(i, j-1) + s('-', y_j) \end{cases} \quad (ii)$$

where $s(a, b)$ is the function that determines the degree of similarity between two alphabet letters a and b. The function is defined as:

$$s(a,b) = \max \begin{cases} 1, & a = b \\ 1, & NE(a) = NE(b) \\ 1, & POS(a) = POS(b) \\ 1 - penalty, & POS(a) \approx POS(b) \\ 0, & a \neq b \end{cases}$$

where NE(a) is the Named Entity (NE) tag of a, and POS(a) is the POS tag of a. If the POS tags of a and b are not the same, but they have a common prefix, the degree of similarity is subtracted with a penalty.

For a sequence X of length n and a sequence Y of length m, totally $(n + 1)*(m + 1)$ scores are calculated by applying the formula (i and ii) recursively. The scores are stored in a matrix $F = F(x_i, y_j)$, and the optimal local alignment can be found by back-tracking in F.

After applying the similarity function to the alignment algorithm, our ABSP generation algorithm extracts general patterns of all three types of tags. We begin by pairing all questions based on their similarity. Closely matched pairs are then aligned and a template that fits both pairs is created. A template (ABSP) is composed of ordered slots, which are chosen according to the corresponding parts of the aligned sentence pair with the following priority: word > NE tag > POS tag. If the sentences for a given slot have nothing in common, the algorithm creates a gap ("－") in that position.

Table 1 shows an aligned pair of sentences. In this case, the algorithm generates the pattern "V － N Na 的 LOCATION Na 是 PERSON," which means a verb followed by a gap, two nouns, a word "的", a location, a noun, a word "是" and a person.. The pattern is generated in this way because, in the first and third positions, the aligned pairs have the same common prefix for POS tag "V, N;" in the second position, they have nothing in common, thus resulting in a generalized gap, "－;" in the fourth and seventh positions, they have the same POS tag "Na;" in the fifth and eighth positions, they have the same words "的, 是;" and in the sixth and ninth positions, they have the same semantic (NE) tag "LOCATION, PERSON. The complete generation algorithm is detailed in Algorithm 1.

Algorithm 1: ABSPs Generation

Input: Question set $S = \{s_1, \ldots, s_n\}$,
Output: A set of uncategorized ABSPs $T = \{t_1, \ldots, t_l\}$.
1: $T = \{\}$;
2: **for** each question s_i from s_1 to s_{n-1} **do**
3:   **for** each question s_j from s_i to s_n **do**
4:     perform ***alignment*** on s_i and s_j, then
5:      pair segments according to similarity matrix F;
6:     generate a common ABSP t from the aligned pairs with the maximum similarity;
7:     $T \leftarrow T \cup t$;
8: **end**;
9: **end**;
10: return T';

Table 2: An example of relation extraction

Questions:	女演員/OCC 蜜拉索維諾/PER 獲得/VJ 奧斯卡/Nb/ORG 最佳/A 女配角/OCC 獎/Na 是/SHI 因/Cbb 哪/Nep 部/Nf 電影/Na
ABSP₁:	*VC Neu **Nb** A **OCC** – Na*
Sentence₁:	...... 而/Cbb **奪得/VC** 一九九五/Neu **奧斯卡/Nb** 最佳/A **女配角/OCC** 的/DE 殊榮/Na …
Relation₁:	{奪得/VC, 奧斯卡/Nb, 女配角/OCC}
ABSP₂:	***PER* *P* *PAR* *ART* *PAR* – *DE Na X* **VJ** ***Nb***
Sentence₂:	…**蜜拉索維諾/PER** 在/O/P/O 「/O/PAR **非強力春藥/ART** 」/PAR 中/Ncd ...... **獲/VJ 奧斯卡/Nb** 獎/Na …
Relation₂:	{蜜拉索維諾/PER, 非強力春藥/ART, 獲/VJ, 奧斯卡/Nb}
Merged Relation:	{ 奪得/VC, 奧斯卡/Nb, 女配角/OCC, 蜜拉索維諾/PER, 非強力春藥/ART, 獲/VJ }

Because sentence alignment is time consuming, we only use the training questions, which consist of 400 NTCIR-5 CLQA questions and 465 questions that we created. For each training question, we retrieve the top 200 most relevant sentences tagged with NEs and POS tags.

2.3. Answer ABSP Selection

To select useful answer ABSPs, we apply the generated ABSPs to the training set. Since a sentence is comprised of several terms, before we can select appropriate ABSPs, we have to filter out redundant words. To achieve this, we first extract important terms from each training question and use them with the answer term to retrieve relevant sentences from the corpus. We define *important terms* as the terms with NEs, the terms with the POS tag 'Nb', and verbs (e.g., 'VC', 'VJ' …). As noted earlier, in our experiment, we only use the top 200 sentences for each question.

The *important terms* are also used in the ABSPs, which have slots for NE/POS tags. The slots are filled with corresponding *important terms*, and the ABSPs are then applied to all the sentences selected for each question. If a one of the ABSPs matches a sentence and the matched segment contains the correct answer to the question, that ABSP is selected as an answer ABSP.

We apply each generated ABSP to its source sentences. When a matched source sentence is found, we extract the corresponding terms from the slots. If the extracted terms do not contain the answer and any of the *important terms* of the source question, the ABSP is removed. In our experiment, we collected 126 useful ABSPs from the 865 training questions. The detail is described in Algorithm 2.

To summarize, for each question, we apply each ABSP to the top 200 sentences selected for each question. If a ABSP matches the sentence and the matched parts of the sentence contain the specified answer, the ABSP is selected as an answer ABSP.

Algorithm 2: ABSPs Selection

Input: A set of ABSPs $T = \{t_1, \ldots, t_k\}$ for selection, the question Q, the answer A, the sentences $S = \{s_1, \ldots, s_n\}$.
Output: Selected set of ABSPs $T' = \{t_1, \ldots, t_l\}$.

```
1:  T' = {};
2:  QTs←extract important terms from Q
3:  for each sentence sᵢ in S do
4:    for each ABSP tⱼ in T do
5:     perform pattern matching on sᵢ with tᵢ, if match then
6:      PTs ←extract terms that match with slots of tⱼ from sᵢ
7:      if PTs contains A and any term in QTs then
8:        T'←T'∪tⱼ ;
9:      end if;
10:   end if;
11:  end;
12: end;
13: return T';
```

2.4. Relation Building and Answer Extraction

To filter candidate answers, we identify relations by matching the sentences selected for a question with the answer ABSPs and then calculate a score for each candidate answer according to these relations.

Given a set of sentences, all the ABSPs are applied to each sentence. If an ABSP matches a sentence, we extract a relation consisting of the matched important terms (i.e., we discard terms that do not have an 'Nb' tag or an NE tag or a verb). New relations are constructed if more than one ABSP matches different terms in a sentence. If the relations contain overlapping terms (i.e., the same term is matched by at least two ABSPs,) we check the inverse document frequency (*idf*) values of those terms. If one *idf* value is higher then a threshold value, the two relations are merged. For example, in Table 1, there are a question, two retrieved sentences from corpus, and two ABSPs that match the two sentences. The first ABSP, ABSP1, extracts Relation1 {奪得/VC, 奧斯卡/Nb, 女配角/OCC} from Sentence1, while ABSP2 extracts the terms「蜜拉索維諾/PER」,「非強力春藥/ART」,「獲/VJ」,「奧斯卡/Nb」 and forms Relation2. Since 「奧斯卡 (Oscurs)」 already exists in Relation1, we examine the *idf* value of 「奧斯卡 (Oscurs)」 and merge it with

Relation1 to form a new one. After all the relations have been constructed for the given question, we use question's *important terms* (女演員 (actress), 蜜拉索維諾 (Mira Sovino), 獲得 (win), 奧斯卡 (Oscurs), 女配角 (Actress in a supporting role), in this example) to calculate the score of relation. A relation score is calculated as the ratio of question's *important terms* to the matched *important terms*. In this case, the number of question's important terms in is 5, and the number of matched *important terms* is 3. Therefore the score of answers belong to this relation is 3/5. For relations that do not have any question's *important term*, we discard the candidate answers contained in them.

After processing all the sentences selected for a question, we rank the candidate answers by the sum of their scores for the sentences in which they appear and retain the top ranked answer(s).

3. Experimental results

We use the development dataset and test dataset of the NTCIR-5 CLQA task and manually create questions for two training datasets, which consist of 400 and 465 questions, respectively. The test dataset, containing 150 questions, was used in the NTCIR-6 CLQA formal test run.

Table 3: Performance in the NTCIR-6 CLQA task

	CC Subtask		EC subtask	
	Precision	Accuracy	Precision	Accuracy
TA	**0.911**	0.340	**0.806**	0.167
ASQA	0.453	0.453	0.280	0.280
TA +ASQA	0.553	**0553**	0.340	**0.340**

Table 4: Performance in the NTCIR-5 CLQA task

	CC Subtask	
	Precision	Accuracy
TA	**0.873**	0.275
ASQA	0.445	**0.445**

Table 2 shows the experimental results for the ABSP approach, the Academia Sinica Question Answering System (ASQA, [7]) without ABSP, and ASQA with ABSP. Since ABSP has higher precision than the pure ASQA system, when applying ABSP to ASQA, we first process questions with ABSP. If ABSP can not find the answer, we then process questions with ASQA. For the CLQA Chinese-Chinese (CC) task, the question coverage was 37.3% and the accuracy when covered (i.e., the precision) was 0.911. Meanwhile, for the English-Chinese (EC) task, the question coverage was 20.7% and the accuracy was 0.807. The accuracy rates were both much higher than the overall accuracy rates, which were 0.553 and 0.34 respectively. The ASQA system with ABSP

won first place in NTCIR-6 CLQA system competition (for both the CC and EC subtasks). Table 3 shows the results of previous competition of ASQA compared to ABSP. The result shows that the performance of ABSP is stable.

4. Discussion and Conclusions

We have proposed a fully automatic alignment-based surface pattern approach that achieves high accuracy in CLQA tasks. Although the coverage of our approach is low, it has achieved acceptable results in both the CC task and the EC task because the accuracy is high enough.

In our research, we found that ABSPs can be improved in several ways. First, ABSP is affected by the word canonicalization problem. For example, "Taiwan" could be "台灣" in simplified form or "臺灣" in traditional form; "thirteen" could be "13" in Arabic numerals, "１３" in capitalized Arabic numerals, "十三" in Chinese numerals, or "拾參" in capitalized Chinese numerals; Foreign person names like Jordan could be "喬丹" or "喬登"; "China" could be "大陸", "中國", and "內地". These canonicalization problems will make ABSP fail to merge some important relations. By applying rules with taxonomy or ontology resources would solve most canonicalization problems.

On the other hand, more accurate semantic tags, which are usually finer-grained, would improve the accuracy while maintaining question coverage. Also, to increase question coverage, in addition to the strategies we adopt, we could also use partial matching because it allows portions of a surface pattern to be unmatched. Allowing overlapping tags is also a possibility, because some errors are caused by tagging, such as wrong word segmentation. However, allowing overlapping tags would need a new alignment algorithm to handle the increased time complexity.

In the EC subtask, we were surprised that the TA algorithm was not overly influenced by the noise introduced by machine translation. We think that, because the technique does not consider the syntax of a question, it can perform well in both mono-language and cross-language situations.

Our most important contribution in this paper is the Alignment-based Surface Pattern (ABSP), which automatically answer relation patterns, and achieves high precision in Cross-Lingual QA system.

5. Acknowledgement

6. References

[1] CKIP, "Autotag," Academia Sinica, 1999.

[2] M.-Y. Day, C.-W. Lee, S.-H. Wu, C.-S. Ong, and W.-L. Hsu, "An Integrated Knowledge-based and Machine Learning Approach for Chinese Question Classification," in *IEEE International Conference on NLPKE*, 2005.

[3] Google, "Google Translate, http://www.google.com/translate_t," 2007.

[4] E. Hovy, U. Hermjakob, and D. Ravichandran, "A Question/Answer Typology with Surface Text Patterns," in *Human Language Technology*, San Diego, CA, 2002.

[5] W.-L. Hsu, S.-H. Wu, and Y.-S. Chen, "Event identification based on the information map-INFOMAP," in *IEEE International Conference on Systems, Man, and Cybernetics*, Tucson, AZ, USA, 2001, pp. 1661-1666.

[6] M. Huang, X. Zhu, Y. Hao, D. G. Payan, K. Qu, and M. Li, "Discovering patterns to extract protein - protein interactions from full texts," *Bioinformatics,* vol. 20, pp. 3604-3612, 2004.

[7] C.-W. Lee, C.-W. Shih, M.-Y. Day, T.-H. Tsai, T.-J. Jiang, C.-W. Wu, C.-L. Sung, Y.-R. Chen, S.-H. Wu, and W.-L. Hsu, "ASQA: Academia Sinica Question Answering System for NTCIR-5 CLQA," in *NTCIR*, Japan, 2005.

[8] Lucene, "The Apache Lucene project, http://lucene.apache.org/," 2007.

[9] I. Muslea, "Extraction Patterns for Information Extraction Tasks: A Survey," in *Workshop on Machine Learning for Information Extraction*, Orlando, 1999.

[10] D. Ravichandran and E. Hovy, "Learning surface text patterns for a Question Answering system," in *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, Philadelphia, Pennsylvania, 2001, pp. 41-47.

[11] M. Saiz-Noeda, A. Su´arez, and M. Palomar, "Semantic Pattern Learning Through Maximum Entropy-based WSD Technique," in *Computational Natural Language Learning (CoNLL-2001)*, Toulouse, France, 2001.

[12] T. F. Smith and M. S. Waternman, "Identification of Common Molecular Subsequences," *Journal of Molecular Biology* vol. 147, pp. 195-197, 1981.

[13] M. M. Soubbotin, "Patterns of Potential Answer Expressions as Clues to the Right Answer," in *Proceedings of the Tenth Text REtrieval Conference (TREC)*, 2001, pp. 175-182.

[14] S. Staab, M. Erdmann, and A. Maedche, "Engineering Ontologies using Semantic Patterns," in *Proceedings of the IJCAI-2001 Workshop on E-Business & Intelligent Web*, Seattle, 2001.

[15] R. T.-H. Tsai, S.-H. Wu, and W.-L. Hsu, "Mencius: A Chinese Named Entity Recognizer Based on a Maximum Entropy Framework," *Computational Linguistics and Chinese Language Processing,* vol. 9, pp. 65-82, 2004.

[16] V. N. Vapnik, *The Nature of Statistical Learning Theory*: Springer, 1995.