

# Spoken Question Answering using Tree-structured Conditional Random Fields and Two-layer Random Walk

Sz-Rung Shiang<sup>1</sup>, Hung-yi Lee<sup>2</sup>, Lin-shan Lee<sup>1</sup>

<sup>1</sup> Graduate Institute of Electrical Engineering, National Taiwan University

<sup>2</sup> Graduate Institute of Communication Engineering, National Taiwan University

r01921050@ntu.edu.tw, t1kagkb93901106@gmail.com, lslee@gate.sinica.edu.tw

## Abstract

In this paper, we consider a spoken question answering (QA) task, in which the questions are in form of speech, while the knowledge source for answers are the webpages (in text) over the Internet to be accessed by an information retrieval engine. Because the recognition results for the spoken questions are less reliable, we use N-best lists in order to have higher probabilities to induce more correct keywords for the questions, but more noisy words are inevitably included as well. We therefore propose a hierarchical labeling method using tree-structured conditional random fields (CRF) to leverage the parse tree information or the syntactic structure obtained from the N-best-lists of the spoken questions, such that the queries for information retrieval can be better formulated. In addition, because queries formulated from the N-best results naturally generate more noisy information, we further propose to use two-layer random walk for re-ranking the retrieved webpages to produce better documents containing answers. Initial experiments performed on a set of question answering pairs collected from quiz shows in Mandarin Chinese verified that improved performance was achievable with the proposed approaches.

**Index Terms:** conditional random fields, question answering, query formulation, random walk.

## 1. Introduction

Most approaches of factoid question answering (QA) [1] tasks are based on information retrieval [2, 3, 4, 5, 6]. Fig.1 shows the flowchart of such approaches. The input questions (transcribed by ASR if spoken) are first processed so as to formulate the queries based on the key terms to be used in retrieving relevant documents or webpages from the Internet or a predefined knowledge source. This approach also needs to detect the answer type [7] out of the questions, such as whether the questions are asking for “location”, “numeric” or “entity”, to find the answers from the passages in the retrieved documents or webpages better matched to the desired answer type. In this paper, we focus on the query formulation, documents retrieval and re-ranking part. To leverage phrases to formulate more meaningful queries and to cope with the less reliable transcriptions, we propose to use tree-structured conditional random fields (CRF) for query formulation [8, 9, 10], and two-layer random walk for re-ranking and selecting better webpages for the questions.

In order to find the proper queries from the questions, sequential labeling is usually applied to decide whether a word (or a phrase) in the question should be considered as query. Hidden Markov models (HMM) [11] and linear-chain conditional random fields (CRF) [12, 13] have been widely used for this purpose, in which the word sequence of the question is used as a sequential input, and the label sequence of whether each word (or phrase) should be taken as query is the

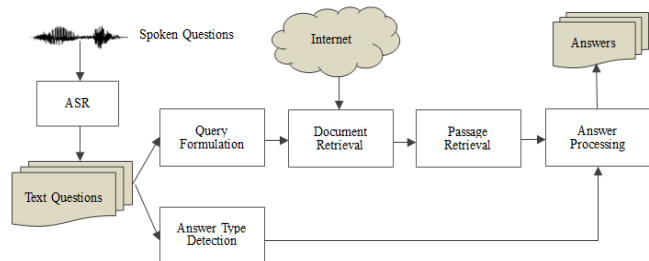


Figure 1: The flowchart of a spoken question answering system based on information retrieval

sequential output. This query formulation problem becomes much more challenging for spoken questions with less reliable transcriptions. In addition, there may be a problem because that some words have the clear meanings, while the phrases combining these words may have the different meanings. We therefore propose to consider N-best transcriptions, take words or phrases rather than words alone as the unit to formulate the queries, and use tree-structured conditional random fields (CRF) [14, 15, 16, 17, 18] with parse trees [19, 20, 21, 22, 23] to leverage the syntactic structures and phrases for this purpose.

In addition, the webpages retrieved by queries formulated with transcriptions of the spoken questions can be very noisy. We therefore propose to use two-layer random walk [24, 25] to re-rank and then select webpages better matched to the spoken questions. Through score propagation over the two graphs on the two different layers, webpages retrieved with incorrect words or phrases may lose their scores, while the scores for those retrieved with correct and important words or phrases may be increased.

Below section II presents query formulation using tree-structured CRF with parse tree structure, while section III illustrates the way two-layer random walk can be used to re-rank the very noisy webpages retrieved with queries from N-best transcriptions. Section IV then demonstrates the experimental setup and results.

## 2. Query Formulation

Sequential labeling is used here to jointly formulate the queries and detect the answer type related words from the N-best transcriptions of the spoken questions. Three labels are used: query term, answer term, and background term. Query term is the word or phrase that we intend to use in the query to retrieve webpages carrying the answer to the question. Those terms labeled as queries are cascaded with “space” as separator to form the new queries. Answer term is the word that reveals the answer type to be used as cues to identify the answer from the retrieved webpages. Background term is the one not important. CRF is an undirected graphical model suitable for sequential labeling work, and have been widely applied on this type of tasks. In the following we present the basic linear-chain CRF,

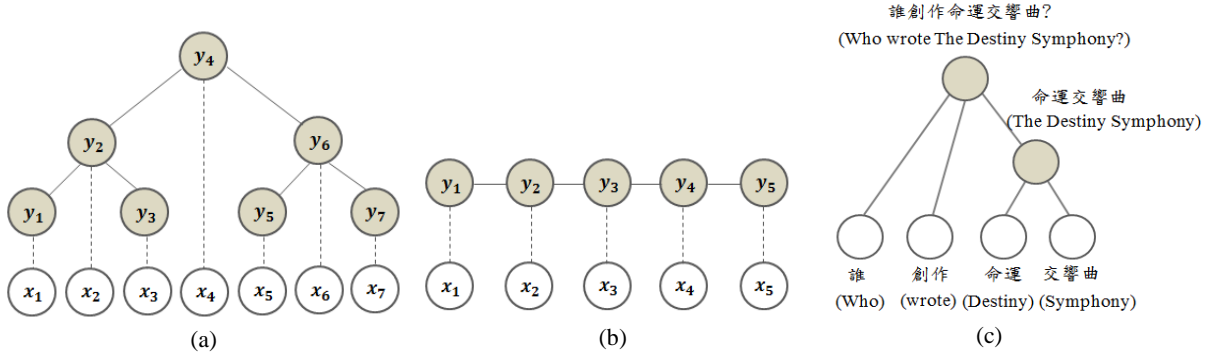


Figure 2: The structures of CRFs: (a) Linear-chain CRF (b) Tree-structured CRF (c) Parse tree of a question.

the proposed tree-structured CRF with parse trees, and the features used in tree-structured CRF.

### 2.1. Linear-chain conditional random fields (CRF)

The widely used linear-chain CRF [12, 13] has nodes in sequential order as in Fig.2(a). Each  $x_i$  here is an input observation represented by a feature vector and  $y_i$  is the corresponding output label for  $x_i$ . The CRF model is to maximize the conditional probability  $P(\mathbf{y}|\mathbf{x})$  as in (1):

$$P(\mathbf{y}|\mathbf{x}) = \frac{1}{Z(\mathbf{x})} \prod_{t=1}^T \phi(\mathbf{y}, \mathbf{x}), \quad (1)$$

where  $\phi(\mathbf{y}, \mathbf{x})$  is the feature function describing the relationships between all labels in the sequence  $\mathbf{y}$  and all observations in the sequence, and  $Z(\mathbf{x})$  is the normalization term. We can further decompose (1) into (2):

$$P(\mathbf{y}|\mathbf{x}) = \frac{1}{Z(\mathbf{x})} \prod_{t=1}^T \exp(\lambda_A \cdot f_A(y_t, y_{t-1}) + \lambda_B \cdot f_B(y_t, \mathbf{x})), \quad (2)$$

where  $f_A(y_t, y_{t-1})$  is the feature vector related to the neighboring labels,  $f_B(y_t, \mathbf{x})$  is the feature vector representing the relationships between the observations and labels, and  $\lambda_A$  and  $\lambda_B$  are the parameters to be learned from the training data.

### 2.2. Tree-structured conditional random fields (CRF)

Linear-chain CRF is not able to model the relationships with hierarchical structure in some tasks. Tree-structured CRF [14, 15, 16, 17, 18] was therefore proposed to solve this problem as in Fig.2(b). Let  $y_p^t$  and  $y_c^t$  be the parent node and child node of  $y_t$  respectively. All the parent-child pairs are considered here in order to model the hierarchical information, and the objective function (2) is reformulated into (3) for tree-structured CRF:

$$P(\mathbf{y}|\mathbf{x}) = \frac{1}{Z(\mathbf{x})} \prod_{t=1}^T \exp(\lambda_e \cdot f_e(y_t, y_p^t, y_c^t) + \lambda_o \cdot f_o(y_t, \mathbf{x})) \quad (3)$$

, where  $f_e(y_t, y_p^t, y_c^t)$  is the feature vector regarding parent-child relationships,  $f_o(y_t, \mathbf{x})$  is the feature vector accounting for relationships between the labels  $\mathbf{y}$  and the observations  $\mathbf{x}$ , and  $\lambda_e$  and  $\lambda_o$  are the parameters to be learned. The parameters  $\lambda_e$  and  $\lambda_o$  can be solved by the Quasi-Newton method such as the L-BFGS algorithm [26], and the decoding of the tree-structured CRF can be achieved by dynamic programming.

### 2.3. Mapping parse trees to tree-structured CRF

the N-best transcriptions for the spoken questions may include recognition errors and noisy words, while a good parser can very often transform them into parse trees based on context-free grammar considering the syntactic structures of the word

sequences as in Fig.2(c). The root node of the tree represents the whole word sequence, while the leaves are all single words. The other nodes illustrate how the word sequence is segmented into smaller syntactic parts hierarchically and thus these nodes denote the phrases with both syntactic and semantic meaning. It is therefore natural to map the nodes in the parse tree in Fig.2(c) to the nodes in the tree-structured CRF in Fig.2(b). As a result, each node in the tree-structured CRF can represent either a word or a phrase, and the tree-structured CRF can be trained and used accordingly.

### 2.4. Features used in tree-structured CRF

Here we present the different types of features used in the tree-structured CRF.

#### 2.4.1. Semantic features

It has been known for long that important words (which tend to be keywords used in queries) usually appear in sentences on small numbers of topics; therefore, topic analysis is useful for the purpose here. Latent dirichlet allocation (LDA) [27] was used in this work for topic analysis. With LDA, for each word  $w_i$  we can infer the topic distribution  $P(T_k|w_i)$ , where  $k \in \{1, \dots, K\}$ ,  $T_k$  is a topic and  $K$  is the total number of topics. We then use the latent topic entropy (LTE) [28] as in (4) to evaluate how a word is concentrated on only a few topics:

$$\text{LTE}(w_i) = -\sum_{k=1}^K P(T_k|w_i) \cdot \log P(T_k|w_i). \quad (4)$$

Larger value of  $\text{LTE}(w_i)$  implies  $w_i$  is more uniformly distributed over all topics and therefore very possibly it is not a keyword. Latent topic significance (LTS) [28] is the measure of how significant a word  $w_i$  is in a topic  $T_k$  as in (5):

$$\text{LTS}(w_i, T_k) = \frac{\sum_{d_j \in D} n(w_i, d_j) P(T_k|w_i)}{\sum_{d_j \in D} n(w_i, d_j) [1 - P(T_k|w_i)]} \quad (5)$$

where  $d_j$  denotes a document in the document set  $D$  considered,  $n(w_i, d_j)$  the occurrence counts of a word  $w_i$  in  $d_j$ . This is actually the ratio of within-topic term frequency to out-of-topic term frequency. Higher value of  $\text{LTS}(w_i, T_k)$  also implies  $w_i$  is more likely to be a key word. For each phrase we computed the mean, minimum and maximum  $\text{LTE}(w_i)$  and  $\text{LTS}(w_i, T_k)$  values of the components words as features.

#### 2.4.2. Parse tree information

From the parse trees, ordered and well-structured probabilistic context-free grammar information regarding the input word sequence can be obtained, including the part-of-speech (POS) tags and semantic roles [29] such as ‘‘head’’, ‘‘quantity’’ and ‘‘time’’, etc. With the semantic role information, the

relationships between constituents may be extracted, and sometimes such relationships represent the core semantic meaning of the word sequence. In the work here, both POS tags and semantic roles of each word or phrase and those of their parents and children in the tree are used as features.

#### 2.4.3. Search engine log data

If a word or a phrase frequently appears in the query logs, we can be more confident that it is important and it can be used as a query; otherwise it may be less important or not a good query. As a result, we take the occurrence counts of a word or a phrase in the query logs to be a feature parameter. We used Sogou search engine click-through log data collected by Sogou.com for this purpose, which contains about 10 M search logs requested by users in three months.

#### 2.4.4. Web-related features

Wikipedia is the free encyclopedia that all users can edit the content and add entries. The titles in the Wikipedia can be considered as publicly acceptable or significant words or phrases. Consequently, we used these titles as a list, and check whether a word or a phrase exists in this list. The data set we used included about 0.7M titles (in Chinese) in the Wikipedia.

#### 2.4.5. Knowledge graph features

E-HowNet [30, 31] is a taxonomy of Chinese words and it contains the knowledge graph of words in a tree structure built by Academia Sinica. From E-Hownet, we can easily find out synonyms describing similar concepts and the relationships with broader or narrower terms. For example, under the node “human” we can find “mathematician”, “doctor”, “woman”, etc. We manually select six primary nodes in E-HowNet and take all the sub-nodes under these six nodes to construct term list of the specific categories. If a word or a phrase includes words in the lists, the value of this feature is 1, and 0 otherwise.

#### 2.4.6. Other features

- TF-IDF: for each word, we calculated the TF-IDF features; while for each phrase we computed the mean, minimum and maximum TF-IDF values of the component words as features.
- Phrase length: number of Chinese characters or English words in the phrase. The phrase length of the parent and child nodes is also included.
- ASR confidence score: For each N-best transcription, the confidence score from language and acoustic models is an important feature. If the confidence score for a word sequence is too low, the word sequence is likely to be wrong and thus no queries should be extracted.
- Question words: if the phrase contains specific question words, such as “who”, “which” or “where”, the value of this feature is 1, and 0 otherwise. This question words set is human-defined and is composed of 9 elements.
- Distance from the nearest question word: the number of steps needed across the parse tree for traversing from one to another.

### 3. Re-ranking for Webpage Selection Using Two-layer Random Walk

Because the webpages are retrieved using the queries generated from N-best transcriptions of the spoken question, they are

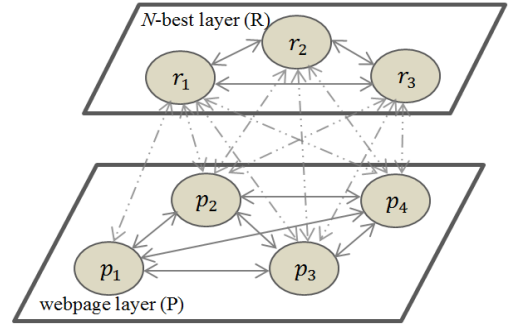


Figure 3: Two-layer random walk framework for webpages re-ranking

naturally very noisy and need to be re-ranked and selected. In this paper, we propose to use two-layer random walk [24, 25] to enhance the scores for the webpages retrieved. The basic idea for score enhancement using random walk is as follows. For a set of objects each with a score, a graph can be constructed with nodes (one node for each object) and edges representing the similarity among these nodes. Because strongly connected (very similar) nodes should have similar scores, the scores of the objects are thus propagated and smoothed over the graph, and enhanced accordingly. For the problems here, we have two different types of objects closely related, the N-best transcriptions and the webpages retrieved; therefore, two-layer random walk can be used to achieve it.

For each spoken question, we build a two-layer graph for re-ranking the scores of all webpages retrieved as shown in Fig.3. This includes an upper layer graph R (N-best layer) for the N-best transcriptions of the question, in which each N-best transcription  $r_i$  is a node ( $r_i \in R$ ); and a lower layer P (webpages layer) for all webpages retrieved, in which each webpage  $p_i$  is a node ( $p_i \in P$ ). Let  $F_R^{(t)}$  be the vector of scores of all N-best transcriptions  $r_i$  on the upper layer R at the  $t$ -th iteration, and  $F_P^{(t)}$  be the vector of the scores of all webpages  $p_i$  on the lower layer P at the  $t$ -th iteration. Let  $E_{RR}$  and  $E_{PP}$  respectively be the edge matrices within the layers R and P, and  $E_{PR}$  and  $E_{RP}$  represent the edge matrices between two layers R and P. The entries in the edge matrices  $E_{RR}$  and  $E_{PP}$  within each layer are the cosine similarity between the unigram vectors using OKAPI/BM25 [32] for the corresponding  $r_i$  and  $p_i$ . The entries in the edge matrices  $E_{PR}$  and  $E_{RP}$  across the layers are simply 1.0 if the corresponding  $p_i$  was retrieved with the queries extracted from  $r_i$ , and 0.0 otherwise. The score propagation over the two-layer graphs is then as in (6):

$$\begin{cases} F_P^{(t+1)} = (1 - \alpha) \cdot F_P^{(0)} + \alpha \cdot E_{PP}^T E_{PR} F_R^{(t)} \\ F_R^{(t+1)} = (1 - \alpha) \cdot F_R^{(0)} + \alpha \cdot E_{RR}^T E_{RP} F_P^{(t)} \end{cases} \quad (6)$$

where  $F_P^{(0)}$  and  $F_R^{(0)}$  are the initial score vectors defined in (6-1) and (6-2), and  $\alpha$  is the interpolation weight belonging to the initial scores and those propagated from other nodes or the other layer.:

$$F_{p_i}^{(0)} = 1/[\text{rank}(p_i) \cdot \text{nbest}(p_i)] \quad (6-1)$$

$$F_{r_i}^{(0)} = \begin{cases} 0.01, & \text{if } \text{webpages}(r_i) = \emptyset \\ 1, & \text{otherwise} \end{cases} \quad (6-2)$$

where  $\text{rank}(p_i)$  is the rank of the webpage  $p_i$  returned by the search engine,  $\text{nbest}(p_i)$  is the order in the N-best transcriptions from which the query retrieving  $p_i$  was extracted, and  $\text{webpages}(r_i)$  is the number of webpages which can be

Table 1. Precision (P) and mean average precision (MAP) of top 3, 5, and 10 returned webpages on manual transcriptions (left) and 5-best ASR results (right), using linear-chain and tree-structured CRF without random walk.

	Manual Transcription		ASR (5-best)	
	(a) Linear CRF	(b) Tree structured CRF	(a) Linear CRF	(b) Tree structured CRF
P@3	0.7362	0.7653	0.4746	0.5138
P@5	0.7221	0.7417	0.4558	0.4793
P@10	0.6669	0.6746	0.4266	0.4333
MAP@3	0.8326	0.8448	0.5716	0.6053
MAP@5	0.8128	0.8392	0.5656	0.5968
MAP@10	0.7832	0.8117	0.5388	0.5636

retrieved by hard matching with the queries of  $r_i$ . Accordingly, the initial scores of the webpages depend on the ranking during retrieval and the order in the N-best transcriptions; besides, if query extracted from an N-best transcription  $r_i$  cannot find any webpage exactly matched, very possibly  $r_i$  includes some recognition errors and the value of  $F_{r_i}^{(0)}$  is reduced to 0.01. It is also noted that all the score vectors  $F_R^{(t)}$ ,  $F_P^{(t)}$  and edge matrices  $E_{RR}$ ,  $E_{PP}$ ,  $E_{PR}$  and  $E_{RP}$  are normalized. By iteratively updating the scores  $F_R^{(t)}$  and  $F_P^{(t)}$  with (6), the scores of objects on each layer can be reinforced by the scores of objects on the other layer. When the score updating converges at iteration T ( $F_R^{(T+1)} \approx F_R^{(T)}$ ,  $F_P^{(T+1)} \approx F_P^{(T)}$ ), a better set of scores for the webpages ( $F_P^{(t)}$ ) can be obtained.

The score propagation within each layer and between the two layers automatically enhances the scores of the webpages in various ways. For example, if a webpage is retrieved by multiple N-best transcriptions, it is more likely to contain the answer so its score should be increased; meanwhile, if the content of a webpage has higher similarity with such webpages with high confidence, it is more likely to contain the answer too. Moreover, if an N-best transcription retrieves more webpages with higher scores, very possibly it is a more accurate transcription, thus we are more confident with the webpages it retrieves. On the other hand, if a webpage is isolated or only weakly connected (low similarity) with other webpages, very possibly it does not include the answer. Similarly, if an N-best transcription retrieves only webpages with lower scores, it may include more recognition errors. All these considerations are automatically taken into account in the two-layer random walk in (6). The retrieved webpages can be finally ranked by the converged scores  $F_P^{(T)}$ , from which we can more likely to find the answer to the question.

## 4. Experiments

### 4.1. Experimental setup

The corpus used in this research included 189 question-answer pairs collected from Chinese quiz shows, in which the answers to the questions are restricted to three types: “human”, “city” and “country”. The spoken questions were produced by a single speaker with a total length of about 58 minutes. ASR gave 12.19% word error rate (WER) and 59.26% sentence error rate (SER) for one-best results and N-best transcriptions were provided. The words and phrases were extracted from the parse tree and those playing the roles of queries or answer words were labeled for each question. We divided this corpus into 3

Table 2. Results of re-ranking the webpages using two-layered random walk

	(a) whole sentence	(b) One-best	(c) 5-best	(d) Re-ranking
P@3	0.4330	0.5021	0.5138	0.5183
P@5	0.4125	0.4770	0.4793	0.5013
P@10	0.3867	0.4367	0.4333	0.4721
MAP@3	0.5153	0.6099	0.6053	0.6239
MAP@5	0.5227	0.5956	0.5968	0.6103
MAP@10	0.5110	0.5631	0.5636	0.5764

folds (63 question-answer pairs for each fold for cross-validation). In each trial, 2 folds (126 pairs) were used as training data and the remaining for testing. All questions were pre-processed by the Chinese word segmentation and parse tree system [22, 23, 33, 34] developed by the CKIP team of Academia Sinica.

We used the precision (P) and mean average precision (MAP) on top 3, 5 and 10 returned search results from Google Search to measure whether the answers of the questions appeared in these webpages. Higher values in these scores imply better question answering systems may be possible.

### 4.2. Experimental results

In Table.1, precision and MAP results of top 3, 5 and 10 retrieved webpages from Google Search based on manual transcriptions (left half) and 5-best ASR results (right half) are listed. In both cases columns (a) shows the results using linear-chain CRF and the columns (b) with the proposed tree-structured CRF. For ASR results on the right, the 5-best transcriptions were used to construct 5 parse trees, with which a forest-structured CRF was used in query formulation. The results in the table were measured on webpages retrieved by all the queries extracted by the forest-structured CRF on the 5-best transcriptions. We can observe that tree-structured CRF obviously outperformed linear-chain CRF for query formulation for both manual transcriptions and 5-best ASR results (column (b) vs (a)). Therefore, using the phrases in the query formulation considering the parse tree information was helpful at least for the corpus considered.

Table.2 compares the precision and MAP on top 3,5,10 webpages with only one-best ASR transcription (column (b)), 5-best transcriptions (column (c)), both without random walk, and those with 5-best transcriptions plus two-layer random walk (column (d)). Column (a) is the results using the whole 5-best ASR transcriptions (without any query extraction technique) as queries to retrieve webpages, and it can be regarded as the lower bound. Column (c) is exactly the right-most column of Table.1, and we see 5-best was mostly better than one-best (column (c) vs (b)). Column (d) verified that the two-layer random walk provided significant improvements over other approaches.

## 5. Conclusions

In this paper, we proposed two new approaches for spoken question answering. This includes leveraging the parse tree information in tree-structured CRF to extract better queries to retrieve webpages through search engine, and using two-layer random walk for re-ranking webpages in order to cope with the noisy the N-best transcriptions. The experimental results reveal improved performance over the baseline approaches.

## 6. References

- [1] Silviu Cucerzan and Eugene Agichtein, Factoid Question Answering over Unstructured and Structure, CIKM 2005.
- [2] Dragomir Radev, Weiguo Fan, Hong Qi, Harris Wu, and Amardeep Grewal, Probabilistic Question Answering on the Web, WWW 2002.
- [3] David Azari, Eric Horvitz, Susan Dumais, and Eric Brill, Web-Based Question Answering: A Decision-Making Perspective, UAI 2003.
- [4] Santosh Kumar Ray, Shailendra Singh, and B.P.Joshi, World Wide Web Based Question Answering System – A Relevance Feedback Framework for Automatic Answer Validation, ICADIWT 2009.
- [5] David Ferrucci, Eric Brown, Jennifer Chu-Carroll, James Fan, David Gondek, Aditya A. Kalyanpur, Adam Lally, J. William Murdock, Eric Nyberg, John Prager, Nico Schlaefler, and Chris Welty, Building Watson: An Overview of the DeepQA Project, AI Magazine 31(3), 59--79, American Association for Artificial Intelligence, 2010.
- [6] Mikhail Ageev, Dmitry Lagun, Eugene Agichtein: The Answer is at your Fingertips: Improving Passage Retrieval for Web Question Answering with Search Behavior Data. EMNLP 2013.
- [7] Kirk Roberts and Andrew Hickl, Scaling Answer Type Detection to Large Hierarchies, LREC 2008.
- [8] Anne Aula, Query Formulation in Web Information Search, WWW 2003.
- [9] Xiangyu Jin, James French, Jonathan Michel, Query Formulation for Information Retrieval by Intelligence Analysts, TR 2005
- [10] Dongyi Guan, Hui Yang and Nazli Goharian, Effective Structured Query Formulation for Session Search, TREC 2012.
- [11] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," Proceedings of the IEEE, vol. 77, no. 2, no. 2, pp. 257–286, 1989.
- [12] Lafferty, J., McCallum, A., Pereira, F, Conditional random fields: Probabilistic models for segmenting and labeling sequence data, *Proc. 18th International Conf. on Machine Learning*, Morgan Kaufmann. pp. 282–289, 2001.
- [13] Sutton, Charles; McCallum, Andrew, An Introduction to Conditional Random Fields, FnTML 2010.
- [14] Joseph K. Bradley and Carlos Guestrin, Learning Tree Conditional Random Fields, ICML 2010.
- [15] Wei Lu, Hwee Tou Ng and Wee Sun Lee, Natural Language Generation with Tree Conditional Random Fields, EMNLP 2009
- [16] Trevor Cohn and Philip Blunsom, Semantic Role Labelling with Tree Conditional Random Fields, CoNLL 2005.
- [17] Jie Tang, Mingcai Hong, Juanzi Li and Bangyong Liang, Tree-structured Conditional Random Fields for Semantic Annotation, ISWC 2006.
- [18] Thomas Mensink, Jakob Verbeek, and Gabriela Csurka, Tree-structured CRF Models for Interactive Image Labeling, IEEE Trans. Pattern Anal. Mach. Intell. 35(2): 476-489, 2013
- [19] Jingjing Liu, Panupong Pasupat, Yining Wang, Scott Cyphers, and Jim Glass, Query Understanding Enhanced by Hierarchical Parsing Structures, ASRU 2013.
- [20] Wang Hongling and Zhou Guodong, Semantic role labeling of Chinese nominal predicates with dependency-driven constituent parse tree structure, Journal of Computer Science and Technology. 28(6): 1117-1126, 2013.
- [21] Li Shoushan, Wang Rongyang and Zhou Guodong, Opinion target extraction via shallow semantic parsing, AAAI 2012.
- [22] Chen Keh-Jiann, Yu-Ming Hsieh, 2004, Chinese Treebanks and Grammar Extraction, Proceedings of IJCNLP-04, pp560-565
- [23] Roger Levy and Christopher D. Manning, Is it harder to parse Chinese, or the Chinese Treebank?, *ACL 2003*, pp. 439-446.
- [24] Yun-Nung Chen, and Florian Metze, Two-Layer Mutually Reinforced Random Walk for Improved Multi-Party Meeting Summarization, SLT 2012
- [25] Sujay Kumar Jauhar, Yun-Nung Chen, and Florian Metze, Prosody-Based Unsupervised Speech Summarization with Two-Layer Mutually Reinforced Random Walk, *IJCNLP 2013*.
- [26] Richard H. Byrd, Jorge Nocedal, and Robert B. Schnabel, Representations of quasi-Newton matrices and their use in limited memory methods, *Math. Program.* 63: 129-156 (1994)
- [27] DM Blei, AY Ng, MI Jordan, Latent dirichlet allocation, the *Journal of machine Learning research* 3, 993-1022.
- [28] Sheng-yi Kong and Lin-shan Lee, "Improved spoken document summarization using probabilistic latent semantic analysis (plsa)," in *Proc. of ICASSP*, 2006.
- [29] You, Jia-Ming, Keh-Jiann Chen, 2004, Automatic Semantic Role Assignment for a Tree Structure, Proceedings of SIGHAN workshop.
- [30] Su-Chu Lin, Shu-Ling Huang, You-Shan Chung and Keh-Jiann Chen, The Lexical Knowledge and semantic representation of E-HowNet, *Contemporary Linguistics*, Vol.15, No.2, pp. 177-194, 2013.
- [31] Wei-Te Chen, Su-Chu Lin, Shu-Ling Huang, You-Shan Chung, and Keh-Jiann Chen, E-HowNet and Automatic Construction of a Lexical Ontology, the 23rd International Conference on Computational Linguistics, Beijing, China, 2010.
- [32] Stephen E. Robertson, Steve Walker, Susan Jones, Micheline Hancock-Beaulieu, and Mike Gatford, Okapi at TREC-3, TREC 1994
- [33] Yu-Ming Hsieh, Ming-Hong Bai, Jason S. Chang and Keh-Jiann Chen, Improving PCFG Chinese Parsing with Context-Dependent Probability Re-estimation, CLP-2012.
- [34] Yu-Ming Hsieh, Duen-Chi Yang, Keh-Jiann Chen, 2005, Linguistically-Motivated Grammar Extraction, Generalization and Adaptation, The Second International Joint Conference on Natural Language Processing (IJCNLP-05).