Jurnal Ilmiah
ILMU KOMPUTER
Universitas Udayana

Vol. XI, No. 1, April 2018 **ISSN 1979 - 5661**

ONTOLOGY-BASED PARAGRAPH EXTRACTION AND CAUSALITY DETECTION-BASED SIMILARITY FOR ANSWERING WHY-QUESTION

A.A.I.N. Eka Karyawati

Computer Science/Informatics Program of Mathematics and Natural Sciences Faculty, Udayana University
eka.karyawati@cs.unud.ac.id

ABSTRACT

Paragraph extraction is a main part of an automatic question answering system, especially in answering why-question. It is because the answer of a why-question usually contained in one paragraph instead of one or two sentences. There have been some researches on paragraph extraction approaches, but there are still few studies focusing on involving the domain ontology as a knowledge base. Most of the paragraph extraction studies used keyword-based method with small portion of semantic approaches. Thus, the question answering system faces a typical problem often occuring in keyword-based method that is word mismatches problem. The main contribution of this research is a paragraph scoring method that incorporates the TFIDF-based and causality-detection-based similarity. This research is a part of the ontology-based why-question answering method, where ontology is used as a knowledge base for each steps of the method including indexing, question analyzing, document retrieval, and paragraph extraction/selection. For measuring the method performance, the evaluations were conducted by comparing the proposed method over two baselines methods that did not use causality-detection-based similarity. The proposed method shown improvements over the baseline methods regarding MRR (95%, 0.82-0.42), P@1 (105%, 0.78-0.38), P@5(91%, 0.88-0.46), Precision (95%, 0.80-0.41), and Recall (66%, 0.88-0.53).

Keyword: Ontology-Based Question Answering, Paragraph Retrieval, Why-Question Answering,Why-Question, Causality Detection

1. INTRODUCTION

In the typical QA systems based on the document collection, the keyword-based approach are usually used to handle each step of the document retrieval process (Soricut & Brill, 2006; Higashinaka & Isozaki, 2008; Mori et al., 2008; Nakakura & Fu-kumoto, 2008; Verberne et al., 2010; Verberne et al., 2011; Oh et al., 2012; Oh et al., 2013). The keyword-based QA provides limited capabilities to capture the conceptualizations associated with user needs and document contents. Thus, the word mismatch often occurs because the query and the documents cannot represent the information correctly.

The word mismatch problem refers to the unsuitable use of words to describe the similar concepts/relations in a question and in documents (i.e., paragraphs). The words used by a user to describe concepts/relations in a question are different from used by authors in documents to describe the same concepts/relation. For example, a question, "Why is a VSM employed in text retrieval system?", and a document that contains multi-word term "vector space model" and relation "in order to", where word "VSM" and multi-word "vector space model" describe the same concept (i.e., "VectorSpaceModel" concept), and question word "why" and relation "in order to" describe the same relation (i.e., "causal" relation). The document even contains concepts and relations asked by the user, is not retrieved as a relevant document. Thus, the word-mismatch problem causes the answers are not accurately extracted because most relevant documents that contain answers will not be retrieved. It will decrease the performance of the why-QA system.

The limitation to capture conceptualization of the user needs and the document contents can be solved by using an idea of semantic-based search that is a searching over the document collection based on meaning rather than a literal string (Fernandez et al., 2011). The previous semantic-based document retrieval methods (Castells et al., 2007; Fernandez et. al., 2011) that employed domain ontology are suitable for general questions, not for the why-questions because the approach does not consider the causality detection.

In order to solve the issue of the word mismatch problem in the keyword-based why-QA, and the issue of regardless of the causality detection in the ontology-based IR, the paragraph extraction method that incorporated the causality detection into the ontology-based IR is proposed.

2. THE PROPOSED PARAGRAPH EXTRACTION METHOD FOR ANSWERING WHY-QUESTION

The proposed ontology-based paragraph extraction method involves two components which are the paragraph filtering and indexing, and the paragraph extraction. The paragraph indexing uses semantic annotations based on the domain ontology underlying the question answering system. In the paragraph extraction, a paragraph scoring method involves two measures including the relevance and the appropriateness measure (Han et al. 2006). Figure 1 presents the graphical representation of the paragraph extraction method. The filled box represents the component that has a contribution. The main contribution is a paragraph scoring method. The proposed paragraph extraction method introduces a scoring formula that incorporates the causality-detection-based similarity (i.e., refers to as the appropriateness measure) into an ontology-based TFIDF model (i.e., refers to as the relevance measure).

As can be seen in Figure 1, a list of the top-10 relevant documents (from document retrieval) is used as input for the paragraph filtering and indexing phase. There are three outputs of this phase which are a list of filtered paragraphs, an inverted semantic index, and a TFIDF matrix. The outputs are used as inputs of the paragraph extraction. The output of the paragraph extraction is a list of extracted paragraphs. CA (a set of causality annotations), OSA (a set of original semantic annotations) and ASA (a set of additional semantic annotations) are semantic annotations of a question obtained from question analysis step (Karyawati et al., 2015).

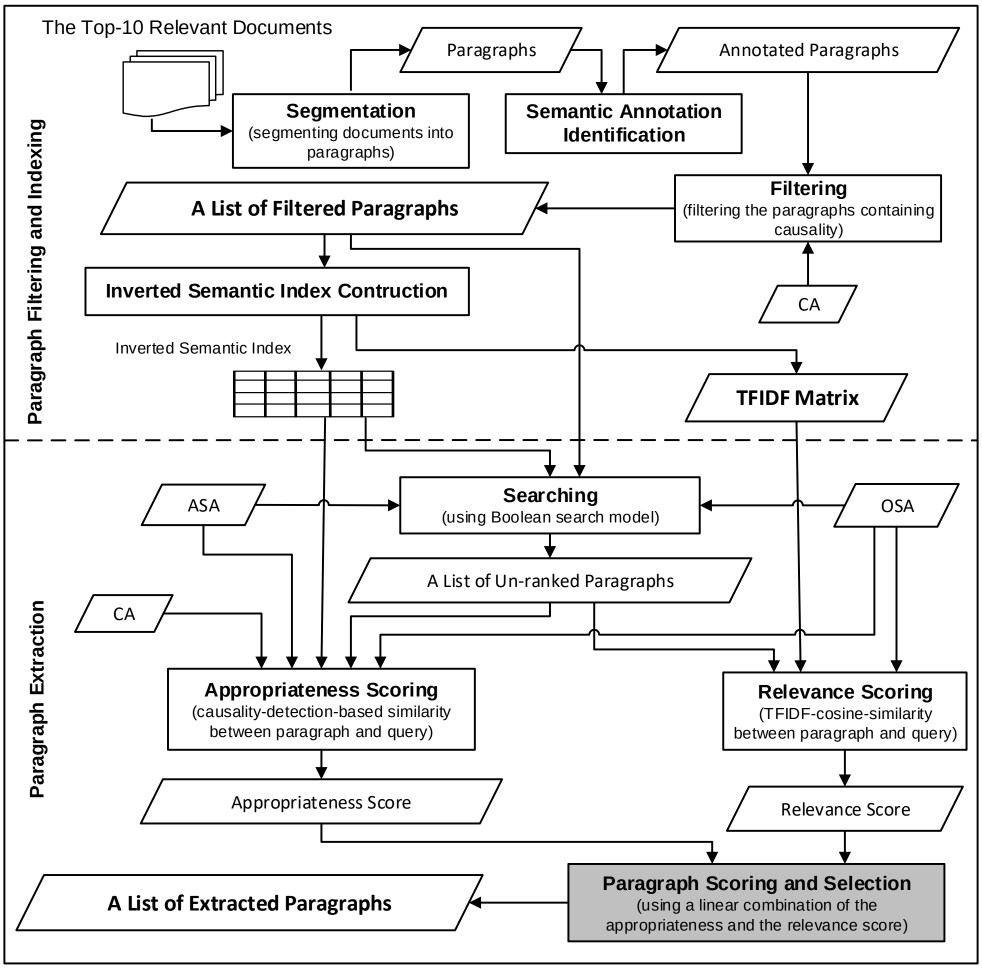
The Top-10 Relevant Documents
Paragraphs
Annotated Paragraphs
Segmentation
(segmenting documents into paragraphs)
Semantic Annotation Identification
Paragraph Filtering and Indexing
A List of Filtered Paragraphs
Filtering
(filtering the paragraphs containing causality)
Inverted Semantic Index Contruction
CA
Inverted Semantic Index
TFIDF Matrix
ASA
Searching
(using Boolean search model)
OSA
A List of Un-ranked Paragraphs
CA
Paragraph Extraction
Appropriateness Scoring
(causality-detection-based similarity between paragraph and query)
Relevance Scoring
(TFIDF-cosine-similarity between paragraph and query)
Appropriateness Score
Relevance Score
A List of Extracted Paragraphs
Paragraph Scoring and Selection
(using a linear combination of the appropriateness and the relevance score)

Figure 1 The Proposed Paragraph Extraction Method

2.1. Paragraph Filtering and Indexing

The paragraph filtering and indexing is performed offline. As can be seen in Figure 1, there are four main components of the paragraph filtering and indexing including the segmentation, the semantic annotations identification, the filtering, and the semantic index construction.

Segmentation

The first component of semantic index construction is segmentation, which has the goal to segment the documents into paragraphs. Verberne's work (Verberne, 2006) resulted that a complete paragraph is a satisfactory answer for more than 80% of why-questions. Thus, the proposed method employs fixed-sized based segmentation by segmenting the retrieved documents (the top-10 documents) into one-paragraph segments.

Semantic annotations identification

After segmentation, a list of semantic annotations of each paragraph is identified based on the *label-instanceName* pairs of the domain ontology-lexicon. Because the paragraphs are usually short texts, the semantic annotations identification do not involve *Lucene* method. A heuristic algorithm is built to identify the semantic entities contained in a paragraph, its occurrences, as well as its positions.

The algorithm is based on n-gram model. Firstly, a paragraph is split into sentences. Then, the longest word sequence (e.g., with length=6) of a sentence is mapped into the semantic entities (i.e., instances in ontology-lexicon), followed by the shorter one (e.g., with length=5), one by one, until no word or word sequence can be mapped. The process is continued until all sentences are handled. The positions of each semantic entity are also recorded.

Filtering

The paragraphs of all documents in the collection are filtered to keep only the paragraphs that contain causalities. This filtering process uses OR-Boolean search, where the Boolean query is given by,

$$Q = X_1 \,\mathrm{OR}\, X_2 \,\mathrm{OR} \ldots \mathrm{OR}\, X_n \tag{1}$$

where X_i is a causality annotation, $X_i \in \{Causality1, Causality2, Causality3, Causality4\}$.

The filtered paragraphs are recorded and saved in text-file. The meta-data of paragraphs includes information of paragraphs ID, paragraph content, a list of semantic annotations (i.e., referring to semantic entities) of the paragraphs, the title of a document where the paragraphs belong to, and semantic annotations of the title.

Inverted Semantic Index Construction

An inverted index of paragraphs collection is constructed for accelerating the paragraph extraction process. In indexing of the paragraphs, besides the occurrence of each term, the positions of the term within a document also store in the index. The reason is that the paragraph scoring also employs a combination of *TFIDF*-based- and causality-detection-based cosine similarity, where the causality detection is estimated by considering the proximity among the semantic annotations of a why-question (i.e., OSA, ASA, CA).

The semantic annotations, calculation of the Semantic Entity Frequency (SEF), and the Semantic Entity Positions (SEP), is performed using a heuristic algorithm. A semantic index of paragraphs collection is constructed after identifying the semantic annotations of each paragraph, and after identifying the semantic entity frequencies and positions within a paragraph. The semantic index is a four-column table. The first column is the instance name, the second column is the paragraph ID, the third column is the SEF, and the fourth column is a list of the SEP within the paragraph.

Inverted indexing process involves two main steps including inverting the semantic index into an inverted semantic index and then sorting the inverted index based on the paragraph ID. Thus, the inverted index is a four-column table. The first column is instance name, the second column is paragraph ID, the third column is SEF, and the fourth column is a list of the SEP within the paragraph. The semantic index construction returns not only an inverted semantic index of the paragraph collection but also a *TFIDF* matrix.

2.2. The proposed paragraph extraction

As can be seen in Figure 1, the paragraph extraction involves four main components which are the searching, the relevance scoring, the appropriateness scoring, and the paragraph ranking and selection.

Searching

Before ranking the paragraphs, the searching is performed to keep only the paragraphs containing all question focuses (got from OSA) and at least one ASA. The reason for using the only question focus not all elements of OSA is that paragraph is usually short. Thus, not all terms (i.e., semantic entities) contain in a question will occur in a relevant paragraph. Similar to the document search, the Boolean search model is applied in this paragraph search. The searched query must satisfy the formula below,

$$Q = \left(X_1 \text{ AND } X_2 \text{ AND} \ldots \text{AND } X_{N_1}\right) \text{AND} \left(Y_1 \text{ OR } Y_2 \text{ OR} \ldots \text{OR } Y_{N_2}\right) \tag{2}$$

where X_1, X_2, …, X_{N1} are elements of question focus, and Y_1, Y_2, …, Y_{N2} are elements of ASA. CA is not involved because the paragraphs have already been filtered using the CA.

Relevance scoring

The relevance score used in this research is the *TFIDF*-based similarity. The *TFIDF*-based ranking of paragraphs refers to cosine similarity score between a paragraph vector and a query vector. The paragraphs vectors are identified from *TFIDF* matrix, where the *TFIDF* matrix is constructed offline in the indexing process. The semantic entity weighting of a query defines TF as 0/1. The TF of a semantic entity is 1 if the semantic entity annotates the question, and otherwise, the value is 0.

A query is represented only by the presence of question focus, instead of the presence of both, OSA and ASA. The reason is that the limitation of the semantic entities contained in a paragraph. The ASA involvement in the query representation will not affect the performance, even decrease it.

The appropriateness scoring

The appropriateness score is estimated by the causality-detection-based similarity value. The causality-detection-based similarity for paragraph ranking is estimated by constructing causality vectors of a question and a paragraph. The causality vectors are constructed on the fly, after identifying semantic annotations of a question (i.e., OSA, ASA, and CA).

To get the most relevant document that contains a suitable causality patterns, the causality-detection based similarity is defined as a maximum value among the causality-based cosine similarity of the four type of causality patterns formulated as,

$$CausalityCosSim = max(CausalityCosSim_1, CausalityCosSim_2, CausalityCosSim_3, CausalityCosSim_4) \tag{3}$$

Where*CausalityCosSim₁*, *Causality-CosSim₂*, *CausalityCosSim₃*, and *CausalityCosSim₄* are the relevance values of a documents (with respect to a query) estimated by calculating the causality-detection-based-similarity corresponding to the causality *Pattern1*, *Pattern2*, *Pattern3*, and *Pattern4*.

Four types of causality patterns based on position/closeness (i.e., proximity) of the terms (i.e., the term proximity considers order of the terms) is (Khoo, 1995; Khoo et al., 2001):

$$\begin{array}{l} Cause - CausalityA - Effect\,(Pattern1) \\ Effect - CausalityB - Cause\,(Pattern2) \\ CausalityA - Effect - Cause\,(Pattern3) \\ CausalityB - Cause - Effect\,(Pattern4) \end{array} \tag{4}$$

The causality vectors are the vectors that represent causality matching values (i.e., 0/1) between the causality patterns that present in the query and that present in the document. Because the causality-detection-based similarity estimates how similar is the causality patterns in a document to the causality patterns in a query, the query causality vectors are set to be the vectors of ones (i.e., vectors whose all elements are 1). Moreover, elements of the document causality vectors are designed to represent the presence of the corresponding causality patterns (of the query) in the document.

The causality-detection-based similarity is a linear combination of OSA-co-occurrence-patterns-based and ASA-co-occurrence-pattern-based cosine similarity. The former are the similarities between the document and the query causality vectors that only consider the co-occurrence of the OSA (i.e., *CausalityCosSimA*). The latter is the similarity between the document and the query causality vector that considers the co-/occurrence of OSA and co-occurrence of ASA in the causality patterns (i.e., *CausalityCosSimB*). The causality cosine similarity of each pattern type (i = 1, 2, 3, 4) is formulated as,

$$CausalityCosSim_i(\boldsymbol{d},\boldsymbol{q}) = \lambda CausalityCosSimA(\boldsymbol{d},\boldsymbol{q}) + (1-\lambda)CausalityCosSimB(\boldsymbol{d},\boldsymbol{q}) \quad (5)$$

where $\lambda \in [0, 1]$.

The paragraph scoring and selection

The proposed scoring formula is defined as a linear combination of the relevance of the paragraph with respect to the query measure and the appropriateness of the writing style measure. The proposed paragraph extraction introduces a scoring method that incorporates the proximity-based causality detection (referring to as an appropriateness measure) and the ontology-based *TFIDF* model (referring to as a relevance measure). The scoring formula is given by,

$$Score\ (\boldsymbol{p},\boldsymbol{q}) = \lambda AppropriatenessScore(\boldsymbol{p},\boldsymbol{q}) + (1\text{-}\lambda)RelevanceScore(\boldsymbol{p},\boldsymbol{q}) \quad (6)$$

where $\lambda \in [0,1]$, and λ is set to be 0.6 because the value is found to work well, empirically. The term **p** and **q** stand for paragraph and question, respectively.

The proposed paragraph selection method uses a specific threshold value to determine whether a sentence is selected or not. The paragraph will be selected if the similarity score is greater than the threshold value. In this research, the threshold value is set to be 0.125, since that value makes the evaluation results seem good.

3. EXPERIMENTS AND RESULTS

3.1. Experiments and Data

The proposed paragraph extraction method is compared against two baseline methods. Both baseline methods are the paragraph extraction that employs a scoring method only based on the relevance measure. The first baseline method uses a relevance measure estimated by using the TFIDF-based similarity using question focuses, where the documents are retrieved using the ontology-based TFIDF method with Query Expansion (QE). The second baseline method uses a relevance measure estimated also by using the TFIDF-based similarity using question focuses, but the documents are retrieved using the ontology-based TFIDF method without QE.

The evaluation is performed by conducting some experiments to measure the effectiveness and efficiency of the methods. The effectiveness of the methods is estimated by calculating the five standard evaluation measures, *MRR* (Mean Reciprocal Rank), *P@1*, *P@5*, *Precision*, and *Recall* of each method (Manning et al., 2008; Baeza-Yates & Ribeiro-Neto, 1999; Thom & Scholer, 2007). Moreover, the efficiency of the methods is estimated by calculating the runtime of the system when the method is executed.

The experiments are conducted by generating randomly 10, 20, 30, and 40 questions from the why-question collection in 10 iterations, where the total number of questions available is 5921 why-questions. The evaluation performances are the average values of each measure.

3.2. Results and Discussion

Table 1 shows the results of the evaluation of the proposed paragraph extraction against the three other methods. Values in bold correspond to the best results for the corresponding metrics. It is the surprising results because the proposed method that only involves *TFIDF*-based similarity in estimating the relevance score outperforms the alternative method that involves the combination of *TFIDF*-based and context-information-based similarity.

As shown in Table 1, the proposed method returns the improvement over the first baseline method in term of MRR (95,2%, 0.82-0.42), *P@1* (105%, 0.78-0.38), *P@5* (91%, 0.88-0.46), *Precision* (95%, 0.80-0.41), and *Recall* (66%, 0.88-0.53), and over the second baseline method in term of *MRR* (173%, 0.82-0.30), *P@1* (200%, 0.78-0.26), *P@5* (144%, 0.88-0.36), *Precision* (208%, 0.80-0.26), and *Recall* (109%, 0.88-0.42).

The second baseline method that is based on *TFIDF*-based similarity without QE becomes the worst method due to the use of only the question focuses without additional semantic annotation in query representation (in retrieving documents). On the other hand, the first baseline method even involves the QE by adding ASA to the original query, but still underperforms the proposed method. The reason is that the first baseline method does not consider the causality detection. Thus, it can be said that the causality detection is important to improve the performance of the paragraph extraction method.

The experiment results show the good values (>=0.78) of all performance measures used in the evaluation of the proposed method because the questions used in the experiments are in well-ordered forms, the question patterns, and the concepts and relations contained in the questions are recognized by the system.

Besides estimating the effectiveness of the proposed methods by comparing the methods based on the five performances metric (i.e., *MRR*, *P@1*, *P@2*, *Precision*, and *Recall* value) as explained above, another aspect also estimated is the efficiency of the methods by comparing the average values of runtimes among the four methods. As can be seen in Table 1, the efficiency of the proposed method is about 5 seconds, the first baseline method is about 2 seconds, and the second baseline method is about 1.5 seconds. It means that the proposed method consumes running time more than twice longer than both baseline methods.

Table 7.3 Comparison results of the proposed paragraph extraction methods against two baseline methods

	Metrics	The Proposed Method	The First Baseline Method	The Second Baseline Method
Data=10	MRR	**0.835646**	0.445972	0.325829
	P@1	**0.788167**	0.413667	0.284917
	P@5	**0.914167**	0.469333	0.369333
	Precision	**0.808244**	0.439192	0.276661
	Recall	**0.914167**	0.587083	0.465833
	RunTime (s)	4.580740	1.878300	1.413460
Data=20	MRR	**0.837527**	0.429016	0.312137
	P@1	**0.787766**	0.393163	0.271144
	P@5	**0.897035**	0.475840	0.380936
	Precision	**0.824419**	0.408427	0.256593
	Recall	**0.898958**	0.528849	0.414186
	RunTime (s)	4.867190	1.930070	1.457500
Data=30	MRR	**0.829875**	0.423656	0.299837
	P@1	**0.795167**	0.382833	0.255611
	P@5	**0.880833**	0.467333	0.363444
	Precision	**0.816830**	0.438579	0.264742
	Recall	**0.880833**	0.547167	0.427444
	RunTime (s)	4.988889	1.989840	1.473040
Data=40	MRR	**0.814366**	0.416073	0.308477
	P@1	**0.779024**	0.376138	0.265282
	P@5	**0.866088**	0.462277	0.376469
	Precision	**0.792932**	0.412638	0.260749
	Recall	**0.869687**	0.534220	0.422969
	RunTime (s)	5.109870	2.039970	1.504400

4. CONCLUSION

By incorporating the causality-detection-based similarity into TFIDF model can improve the performance of paragraph extraction in answering why-questions. The proposed ontology-based paragraph extraction showed the significant improvement over the ontology-based method without causality detection. The proposed method shows the improvements regarding MRR (95%, 0.82-0.42), P@1 (105%, 0.78-0.38), P@5(91%, 0.88-0.46), Precision (95%, 0.80-0.41), and Recall (66%, 0.88-0.53).

REFERENCES

[1] Baeza-Yates, R. and Ribeiro-Neto, B., 1999, *Modern Information Retrieval*, ACM Press, New York.

[2] Castells, P., Fernández, M. and Vallet, D., 2007, An adaptation of the vector space model for ontology-based information retrieval, *IEEE Transactions on Knowledge and Data Engineering*, Vol. 19, No. 2, pp. 261–272.

[3] Fernández, M., Cantador, I., López, V., Vallet, D., Castells, P. and Motta, E., 2011, Semantically enhanced Information Retrieval: An ontology-based approach, Web Semantics: Science, Services and Agents on the World Wide Web, Vol. 9, pp. 434–452.

[4] Han, K.-S., Song, Y.-I., and Rim, H.-C., 2006, Probabilistic model for definitional question answering, *In SIGIR '06: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pp. 212–219.

[5] Higashinaka, R. and Isozaki, H., 2008, Corpus-based question answering for why-questions. *In Proceedings of IJCNLP*, Hyderabad.

[6] Karyawati, A.A.I.N.E., Winarko, E., Azhari, & Harjoko, A., 2015, Ontology-based Why-Question Analysis Using Lexico-Syntactic Patterns, *International Journal of Electrical and Computer Engineering (IJECE)*, Vol. 5, No. 2, pp. 318-332.

[7] Khoo, C. S.-G., 1995, Automatic Identification of Causal Relations in Text and their use for Improving Precision in Information Retrieval, *Ph.D. Dissertation*, Information Transfer in the School of Information Studies, Syracuse University, Syracuse, NY.

[8] Khoo, C. S. G., Myaeng, S. H. and Oddy, R. N., 2001, Using Cause-Effect Relations in Text to Improve Information Retrieval Precision, *Journal of Information Processing and Management*, Vol. 37, pp. 119-145.

[9] Manning, C. D., Raghavan, P. and Schutze, H., 2008, Introduction to Information Retrieval, Cambridge University Press, New York.

[10] Mori, T., Sato, M. and Ishioroshi, M., 2008, Answering any class of Japanese non-factoid question by using the Web and example Q&A pairs from a social Q&A website, *In IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, Sydney, Australia.

[11] Nakakura, S. and Fukumoto, J., 2008, Question Answering System beyond Factoid Type Questions, *In the 23rd International Technical Conference Circuits/Systems (ITC-CSCC 2008)*, Yamaguchi, Japan.

[12] Oh, J.-H., Torisawa, K., Hashimoto, C., Kawada, T., De Saeger, S., Kazama, J., and Wan, Y., 2012, Why Question Answering using Sentiment Analysis and Word Classes, *In Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural language Learning*, Jeju Island, Korea.

[13] Oh J.-H, Torisawa K, Hashimoto C, Sano M, Saeger SD, Ohtake K., 2013, Why-Question Answering using Intra- and Inter-Sentential Causal Relations, *In Proceeding of the 51st Annual Meeting of the Association for Computational Linguistics*, Bulgaria.

[14] Soricut, R. and Brill, E., 2006, Automatic Question Answering Using the Web: Beyond the Factoid. *Journal of Information Retrieval - Special Issue on Web Information Retrieval*, Vol. 9, No. 2, pp. 191–206.

[15] Thom, J.A. and Scholer, F., 2007, A Comparison of Evaluation Measures Given How Users Perform on Search Tasks, *In Proceedings of the 12th Australasian Document Computing Symposium*, Melbourne, Australia.

[16] Verberne, S., 2006, Developing an Approach for Why-question answering, *In Conference Companion of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2006)*, Trento.

[17] Verberne, S., Boves, L., Oostdijk, N., and Coppen, P., 2010, What is not in the bag of words for why-QA?, *Computational Linguistics*, Vol. 32, No. 2, 229–245.

[18] Verberne, S., Boves, L, and Kraaij, W., 2011, Bringing Why-QA to Web Search. *In Proceedings of ECIR ’11*, Dublin, Ireland.