Restricting Answer Candidates Based on Taxonomic Relatedness of Integrated Lexical Knowledge Base in Question Answering

Jeong Heo, Hyung-Jik Lee, Ji-Hyun Wang, Yong-Jin Bae, Hyun-Ki Kim, and Cheol-Young Ock

This paper proposes an approach using taxonomic relatedness for answer-type recognition and type coercion in a question-answering system. We introduce a question analysis method for a lexical answer type (LAT) and semantic answer type (SAT) and describe the construction of a taxonomy linking them. We also analyze the effectiveness of type coercion based on the taxonomic relatedness of both ATs. Compared with the rule-based approach of IBM's Watson, our LAT detector, which combines rule-based and machine-learning approaches, achieves an 11.04% recall improvement without a sharp decline in precision. Our SAT classifier with a relatedness-based validation method achieves a precision of 73.55%. For type coercion using the taxonomic relatedness between both ATs and answer candidates, we construct an answer-type taxonomy that has a semantic relationship between the two ATs. In this paper, we introduce how to link heterogeneous lexical knowledge bases. We propose three strategies for type coercion based on the relatedness between the two ATs and answer candidates in this taxonomy. Finally, we demonstrate that this combination of individual type coercion creates a synergistic effect.

Keywords: Question Answering, Answer Type, Type Coercion, Taxonomy, Taxonomic Relatedness.

Manuscript received Aug. 12, 2016; revised Jan. 2, 2017; accepted Jan. 4, 2017. This work was supported by Institute for Information & Communications Technology Promotion (IITP) grant funded by the Korean Government (MSIP) (No. R0101-16-0062, Development of Knowledge Evolutionary WiseQA Platform Technology for Human Knowledge Augmented Services).

Jeong Heo (corresponding author, jeonghur@etri.re.kr), Hyung-Jik Lee (leehj@etri.re.kr), Ji-Hyun Wang (jhwang@etri.re.kr), Yong-Jin Bae (yongjin@etri.re.kr), and Hyun-Ki Kim (hkk@etri.re.kr) are with the SW & Contents Research Laboratory, ETRI, Daejeon, Rep. of Korea.

Cheol-Young Ock (ocky@ulsan.ac.kr) is with the School of IT Convergence, University of Ulsan, Rep. of Korea.

I. Introduction

With the exponential growth of information on the Web and the advent of mobile environments, the importance of question answering (QA) has increased. Traditional QA systems generally consist of four components: question analysis, information retrieval, answer candidate extraction, and candidate ranking [1]–[5]. A core function of question analysis is to determine what the question is asking. This is generally called the *expected answer type* (*AT*). ATs can be divided into two categories. First, ATs can be categorized according to semantic classes predefined by a statistical analysis of question types (QTs), which we refer to as a *semantic answer type* (*SAT*). These are called *type-and-generate approaches* because only entities having a type related to the SAT are generated as answer candidates. The general processing flow of this approach is depicted in Fig. 1(a). Second, ATs can be categorized into concept words that impose a constraint on the type of answer, which we refer to as a *lexical answer type* (*LAT*), based on the IBM Watson system [6]–[8]. Answer candidates are generated using multiple methods without reference to the LATs [8]. Then, a type coercion between LATs and answer candidates is conducted using lexical KBs such as WordNet and Yet Another Great Ontology. The results of the type coercion are used as one of the features for ranking [8]. For this reason, these approaches are called *generate-and-type*, as shown in Fig. 1(b).

We present examples to understand the difference of two ATs.

Q1: Who killed John F. Kennedy? (Answer: Lee Harvey Oswald)
Q2: The ratio of a distance on a map to the corresponding actual distance. (Answer: scale)

One difference between these two approaches is whether the AT is at a lexical or semantic level. There is no LAT in Q1, but

https://doi.org/10.4218/etrij.17.2816.0066

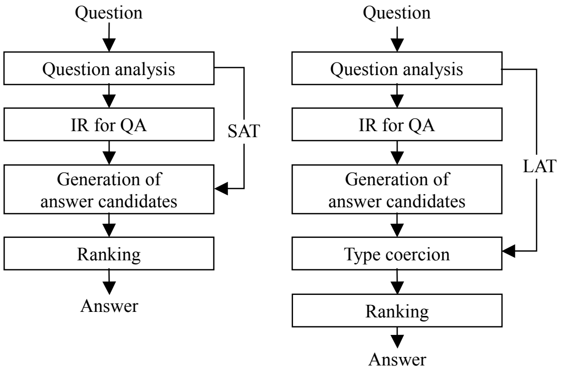

Fig. 1. Approaches for QA: (a) type-and-generate approach and (b) generate-and-type approach.

the SAT of Q1 is PERSON because of the interrogative "who," whereas in Q2, the SAT is uncertain, but the LAT "ratio" can be detected. PERSON and "ratio" can influence the answers to Q1 and Q2, respectively. Because "Lee Harvey Oswald" is recognized as a PERSON by the named entity (NE) recognizer, we can restrict the answer candidates for Q1. In Q2, "ratio" and "scale" have a hypernym/hyponym relationship in WordNet ("ratio/quotient/proposition/scale").

A type-and-generate approach suffers from a dependence on the semantic classes of the SATs. A drawback of a generate-and-type approach is excessive computations owing to the type coercion between the LATs and answer candidates. In addition, questions without an LAT are not the subject of type coercion based on this AT. IBM's Watson team announced that 15% of the *Jeopardy!* questions did not explicitly assert an LAT [8].

To effectively use two approaches that are complementary, we propose an approach using the taxonomic relatedness of two ATs for a type coercion. In addition, we propose a hybrid method for LAT detection and SAT classification and describe how to construct the taxonomy linked between these two ATs. An LAT detector, which is a hybrid of a rule model and machine-learning (ML) model, showed a high recall compared to a rule only-based model, which applies to IBM's Watson. To improve the SAT classification, we used the relatedness to LAT on the taxonomy we constructed. We compared various distance similarities to our proposed similarity as the relatedness measure. The proposed similarity showed the best performance. We propose an approach using both types of AT in the type coercion. We understand that this approach has a synergistic effect.

The remainder of this paper is organized as follows. In Section II, we review related work in this area. In Section III, we propose a method for recognizing two ATs and introduce the method for constructing a taxonomy using heterogeneous lexical KBs. In addition, we describe the type coercion using the LAT and SAT in detail. In Section IV, we analyze the results of our experiment. Finally, in Section V, we summarize our contributions.

II. Related Work

The QA competition in TREC has played a leading role in improving QA technologies since 1999. The degree of difficulty of QA has gradually increased at these QA competitions (for example, from a factoid QA to a list QA, definition QA, interactive QA, and live QA) [1].

The question-processing module of the LASSO system involves four major steps [3]. First, the QT is determined based on 5W1H (why, what, where, when, who, and how). Second, the AT that represents the semantic category of the expected answer is recognized. The AT of a "who" question is clearly PERSON, but the AT of a "what" question is ambiguous. Thus, the LASSO system introduced the so-called "*question focus*" (*QF*) concept to resolve this ambiguity. The detection of the QF is the third step. Finally, the question-processing module reformulates queries from a question for the search engine. LASSO system has the major drawback of a type-and-generate approach because it uses an SAT. To address this problem, taxonomic studies on a fine-grained SAT using lexical knowledge bases (KBs) have been proposed [9].

In [9], an answer-type taxonomy for an open-domain QA was introduced. Three steps were proposed for building the taxonomy. In the first step, the most representative nodes are manually added at the top of the taxonomy. The second step forms a many-to-many mapping between the NE categories and the top of the taxonomy. In the final step, each leaf of the top added into the taxonomy is manually linked to one or more subhierarchies from WordNet. The expected AT is determined based on syntactic parses.

After IBM's Deep Blue chess computer was revealed in 1996, IBM selected a QA as a new area of challenge for artificial intelligence and unveiled Watson, a system capable of answering questions developed by IBM's DeepQA project [6]. Watson was specifically developed to answer questions on the quiz show *Jeopardy!*, which is a well-known syndicated US TV quiz show that has been on the air since 1984. On January 14, 2011, Watson beat the two best *Jeopardy!* champions in a competition. This historical event led to a revival of artificial intelligence research.

Watson was developed based on IBM's prior QA system called *PIQUANT* [2]. *PIQUANT*, which has scored in the top tier of systems in TREC evaluations, presumes a static predetermined set of ATs. However, owing to the breadth of domains and the complexity of language, a new definition of an AT was required.

https://doi.org/10.4218/etrij.17.2816.0066

To overcome the limitations of a static predetermined set of ATs, IBM's Watson used an LAT rather than an SAT. To determine the LAT, Watson detected QFs using several patterns. The headword of the QF is generally chosen as the LAT. The detected LATs have a confidence value produced by the logistic regression classifier. LATs with a low confidence value are filtered to improve the precision [7]. As mentioned above, the LASSO system analyzes questions for three factors, that is, 5W1H-based QTs, QFs, and the semantic categories of the AT. Watson detects the QF based on patterns, and choses the headword of the QF as the LAT.

We propose a QA system that combines the strong points of LASSO with the merits of Watson. We call this QA system *WiseQA*. *WiseQA* recognizes an SAT that is similar to the AT of LASSO and detects the LAT just as Watson does. To improve the recall of the LAT, we combined an ML model based on a sequence-labeling algorithm with a rule-based approach. In addition, we propose an SAT classifier based on an ML approach and lexico-semantic rules. Furthermore, we construct an answer-type taxonomy linking several heterogeneous lexical KBs, such as the NE categories, WordNet, and Wikipedia categories.

III. WiseQA

As shown in Fig. 2, *WiseQA* uses five steps to answer questions. The core topic of this paper is the effectiveness of type coercion using the taxonomic relatedness between both an LAT and SAT. For this purpose, we describe the components related to a type coercion in detail below.

1. Question Analysis

In the question analysis component, a QF and an LAT are detected from words within questions, and an SAT is classified into a predefined semantic category. The question analysis component employs various linguistic analysis techniques such as part-of-speech (POS) tagging, chunking, NE tagging [10], word sense disambiguation, dependency parsing, semantic role labeling (SRL) [11], and co-reference resolution [12].

A. Detection of QF and LAT

In Watson, the QF is the part of the question that refers to the answer [7]. The targets of the QF in Watson are a noun phrase with a determiner "this" or "these" and some pronouns. We extended the target of the QF to include interrogative words, which are useful clues for understanding what a question is asking. For the QF detection, we added the following patterns to Watson:

- An interrogative pronoun and interrogative adverb.

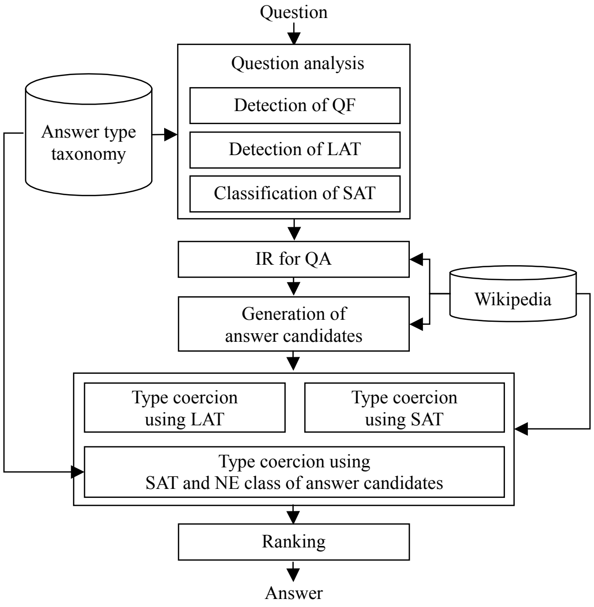

Fig. 2. Overview of WiseQA.

- A noun phrase with interrogative adjectives.

In the second pattern, which includes "What X is Y?" and "Which X is Y?" X is the LAT. For example, the LAT of the question, "What movie won the Oscar in 2014?" is "movie."

We filtered out incorrect QFs using the co-reference resolution component. In Q3, which is an example of QF filtering, "this man," "what," and "this cipher" were detected as QF candidates based on our patterns. However, because "this man" refers to "Alan Turing," this particular QF candidate was filtered out.

Q3: Alan Turing is considered to be the father of computer science and artificial intelligence. During the Second World War, this man devised a number of techniques for breaking German ciphers. What is this cipher? (Answer: Enigma)

The LAT has a semantic relationship with an expected answer such as instance-of, part-of, and is-a. According to their semantic relationships, the LAT can limit the answer candidates based on the taxonomy.

For the detection of the LAT, we chose the headword of the QF, as did Watson. However, like Q2, many questions are an incomplete sentence (such as a phrase) without a matching pattern. This problem causes a low recall in LAT detection. To improve the recall of LAT detection, we applied ML technology. We treated LAT detection as a sequence-labeling problem. We limited the target of the LAT to a noun because most of the words in a taxonomy are a noun or noun phrase. Figure 3 shows an example of noun-based sequence labeling. For the learning of the LAT, we extracted the following features from the question:

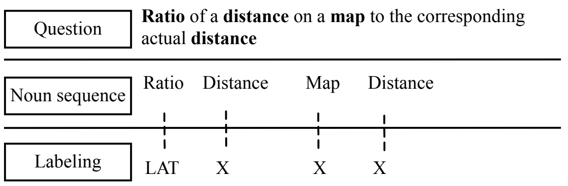
Question
Ratio of a distance on a map to the corresponding actual distance
Noun sequence
Ratio
Distance
Map
Distance
Labeling
LAT
X
X
X

Fig. 3. Example of noun-based sequence labeling for LAT detection.

- Position: The position of the word in the sentence.
- Morpheme: The POS tag, morpheme, bi-grams of the morpheme, and information of adjacent morphemes.
- NE: An NE class of the word.
- Chunk: A label of the chunk containing the word.
- Dependency parsing: A label of the phrase containing the word, a label of the modifier, a string of the modifier, and the morpheme feature of the modifier.
- SRL: A semantic role label of the SPO (subject–predicate–object) related to the word.
- QT: An interrogative type in the question.
- QF: The Boolean value (true/false) according to whether a QF is in question.
- W2V: The Word2Vec of words in the question [13].

LATs detected by the two models are joined together as (1). LATs overlapping the $LAT_{\text{PATTERN}}(q)$ and $LAT_{\text{ML}}(q)$ have a high confidence value.

$$LAT_{\text{TOTAL}}(q) = LAT_{\text{PATTERN}}(q) \cup LAT_{\text{ML}}(q). \tag{1}$$

B. Classification of SAT

We used the extended-NE category of [10] for the SAT. The extended-NE category consists of a hierarchical structure with four depths and 183 categories. To classify the SAT of the question, we also combined the ML approach with the lexico-semantic rule approach. In an ML approach, SAT recognition is a multiclass classification problem. Figure 4 shows the flow of SAT classification based on a ML approach. A hybrid between the two approaches is represented through a linear combination, as in (2).

$$SAT(q) = \text{argmax } w_k\left(\left(ML_k(q)\times\alpha\right)+\left(R_k(q)\times\beta\right)+\gamma\right), \quad k \in 1, \ldots, K, \;\; \alpha+\beta+\gamma = 1, \tag{2}$$

where k is a class of the SAT, q is an unseen question, and $ML_k(q)$ and $R_k(q)$ are the confidence values of class k returned by the classifiers based on the ML and rules, respectively. The confidence value of each rule depends on the precision of the training data. In addition, γ is the weight for class k overlapping between the two methods. If class k does not overlap, γ is 0. The weights of α, β, and γ are empirically determined. In *WiseQA*, α, β, and γ are 0.45, 0.45, and 0.1, respectively. We used the following features for an ML-based classifier.

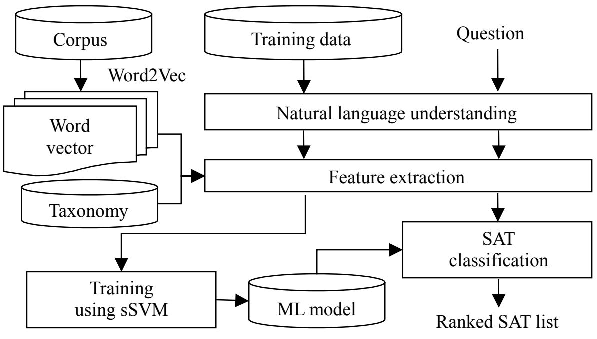

Fig. 4. Flow of SAT classification based on ML approach.

- Morpheme: Bi-grams of the morpheme.
- NE: An NE class of the word.
- Noun phrase: The first noun phrase and the last noun phrase in question.
- Chunk: A label of the chunk containing the word and bi-grams of the word.
- Dependency parsing: A label of the phrase containing the word, a label of the modifier, a string of the modifier, and the morpheme feature of the modifier.
- SRL: A semantic role label of the SPO related to the word.
- QT: An interrogative type in the question and information of adjacent noun phrases.
- QF: The QF, bi-grams of the QF, words adjacent to the QF, and bi-grams of words adjacent to the QF.
- LAT: The LAT and bigrams of the LAT.
- W2V: The Word2Vec of the word in the question [13], [14].
- Concept: An abstract concept of the word in the question. The abstract concept of a word is the hypernym on the predefined depth from the root on the taxonomy.

2. Construction of Answer-Type Taxonomy

Lexical KBs are the core resources in several research fields, including natural language processing, information retrieval, the semantic Web, and QA. Many studies have been carried out exploring the importance of lexical KBs, and several methodologies have been developed for the purpose of linking WordNet to Wikipedia [15], [16].

To describe the rich relationship between the LAT and SAT, *WiseQA* requires a useful taxonomy with links among several heterogeneous KBs. For this purpose, we chose four lexical KBs as follows.

- UWordMap: Korean word map constructed by the University of Ulsan, Rep. of Korea [17].
- KorLex: Korean WordNet, translated by the University of Pusan, Rep. of Korea [18].
- NE classes: NE classes categorized by ETRI, Rep. of Korea [10].

- Korean Wikipedia: Categories of the Korean Wikipedia.

Words within the four lexical KBs are interconnected with a semantic relationship for an efficient type coercion between two ATs and the answer candidates.

A. Linking UWordMap and KorLex

The definition (or glossary) of words within UWordMap and KorLex is borrowed from a standard Korean dictionary, and thus we can automatically link KorLex to UWordMap If word and sense code within KorLex are identical to those of UWordMap and the definitions of two words are same, these words can be linked between KorLex and UWordMap. Information about the linking between two lexical knowledges is shown in Table 1. We named the integrated lexical KB *WiseWordNet*.

B. Linking WiseWordNet and NE Classes

Each of the NE classes is manually connected to a concept node (or synset) on *WiseWordNet*. The node linked to the NE class should have instances with the same NE class. Nodes without a connection inherit the NE class of the upper node based on the least-upper-bound. For example, "college" should only be linked to OGG_EDUCATION in Fig. 5. However, OG_OTHERS linked to "organization" cannot be inherited by "college" because of the violation of the least-upper-bound property. Nodes can be linked from multiple NE classes.

Table 1. Statistical information about linking KorLex to UWordMap.

Unit	Noun	Verb	Adjective	Adverb	Total
# of word sense	121,013	20,270	51,788	8,931	202,002
# of mapping	69,429	9,981	17,663	2,914	99,987
Ratio (%)	57.37	49.24	34.11	32.63	49.50

Information for *WiseWordNet* is shown in Table 2. KorLex has a synset because it is translated from WordNet. However, the node of UWordMap is not a synset, but a concept word.

We verified the completeness of the linkage using two methods. One was to do a crosscheck on the links by hand. The crosscheck process was conducted by two groups. Each group was organized by two experts with a master's degree in linguistics. The other way was to verify the links using evaluation data. For the evaluation, we used a manually annotated set of 2,852 questions chosen from *JangHak Quiz*. *JangHak Quiz* is a Korean TV show in which high school students answer questions. The questions were tagged with the LATs and SATs.

We present the Q4 to understand how to automatically verify the linking completeness.

Q4. This bank was a financial institution created in Florence during the 15th century. The founding family of this bank fostered and inspired the birth of the Italian Renaissance. What is this bank? (Answer: Medici bank)

In Q4, the LATs are "bank" and "institution," and the SAT is OGG_ECONOMY. We generated LAT-SAT pairs such as "bank"-OGG_ECONOMY and "institution"-OGG_ECONOMY. To verify these pairs, we checked whether the LAT-SAT pairs were connected. Table 3 shows the results of the verification. The agreement ratio is the degree of consensus between the two groups.

Table 2. Information of *WiseWordNet*.

KBs	# of noun words	# of nodes (or synsets)	# of nodes (or synset) directly linked to NE classes (ratio)
KorLex	121,013	101,867	7,206 (7.1%)
UWordMap	362,960	362,960	10,140 (2.8%)

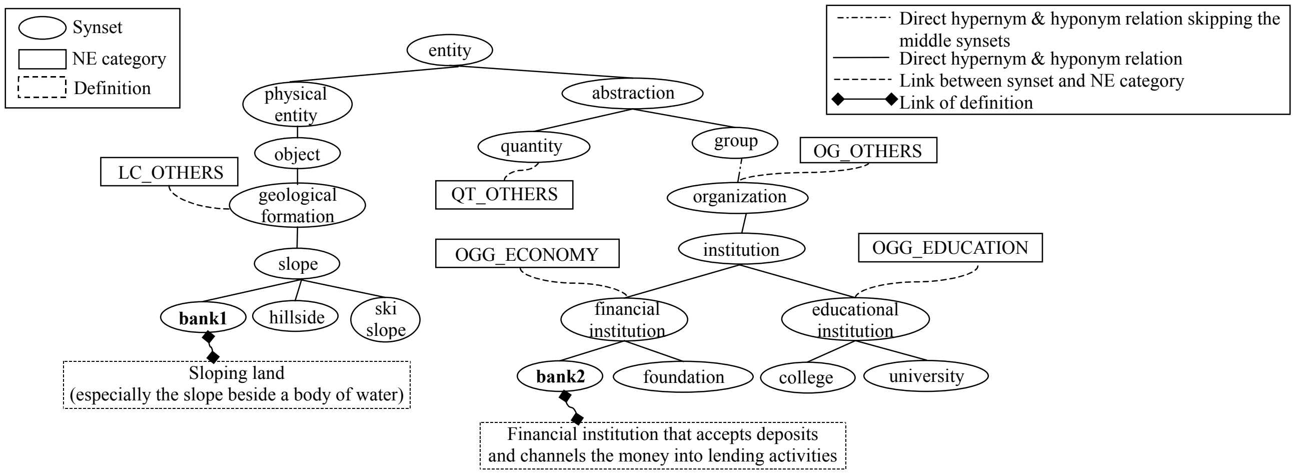

Fig. 5. Example of linking WiseWordNet to NE classes.

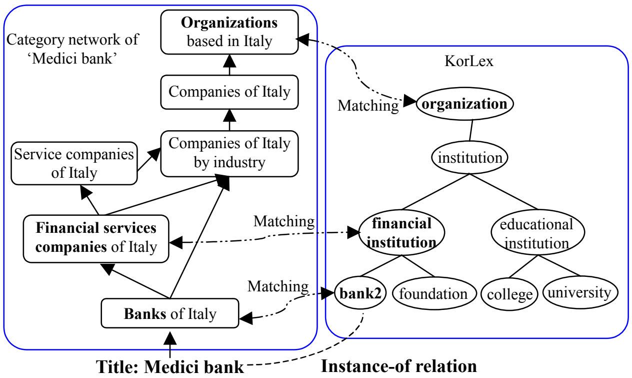

Fig. 6. Example of linking between WiseWordNet and Wikipedia titles.

Table 3. Verification results of linking completeness.

KBs	Agreement ratio of manual cross-check (%)	Ratio of LAT-SAT pairs automatically connected (%)
KorLex	84	89
UWordMap	91	86

C. Linking WiseWordNet and Wikipedia Titles

In [19], the vast majority of *Jeopardy!* answers were shown to be titles of Wikipedia articles. For the QA performance, it is quite significant that Wikipedia titles be linked to *WiseWordNet*. For this purpose, we exploited the structural similarity between nodes on *WiseWordNet* and the categories of the Wikipedia titles. "Bank," the headword of "Medic bank," is ambiguous, as shown in Fig. 5. Linking disambiguation is resolved through three steps. First, we extracted the headwords of the Wikipedia categories and constructed a network of categories. Second, we extracted the hypernym path of the ambiguous word on *WiseWordNet*. Finally, we calculated the structural similarity between the network of categories and the hypernym path of the ambiguous word. The structural similarity is based on the ratio of matching words between the bag-of-words that is extracted from the network of categories and the hypernym path in the *WiseWordNet*. As shown through the example in Fig. 6, three categories of "Medici bank" are matched hypernyms of "bank2." Thus, "Medici bank" is linked to "bank2" with the instance-of relation.

3. Measures of Taxonomic Relatedness.

We adopted the following distance similarity algorithms as a taxonomic relatedness measure.

Hirst and St-Onge: Two lexicalized concepts are semantically close if their WordNet synsets are connected by a path that is not too long and that does not change direction too often [20]. The distance similarity is given by

$$DistSim_{\mathrm{hso}}(c_1,c_2) = C - len(c_1,c_2) - \left(k \times turns(c_1,c_2)\right), \quad (3)$$

where $len(c_1, c_2)$ is the length of the shortest path between two concepts, and $turns(c_1, c_2)$ is the number of changes of direction in the path. Both C and k are constants.

Leacock and Chodorow: The distance similarity is given by

$$DistSim_{\mathrm{lch}}(c_1,\ c_2) = -\log\left(len(c_1,c_2)/(2 \times D)\right), \quad (4)$$

where D is the maximum depth of the taxonomy [21].

Wu and Palmer: This measure calculates the relatedness by considering the depths of the two synsets in WordNet, along with the depth of the lowest common subsumer [22]. This measure is given by

$$DistSim_{\mathrm{wpa}}(c_1,c_2) = \frac{2 \times LCS(c_1,c_2)}{Depth(c_1) + Depth(c_2)}, \quad (5)$$

where $LCS(c_1, c_2)$ is lowest common subsumer between c_1 and c_2, and $Depth(c_1)$ is the depth of c_1 in the WordNet taxonomy.

Resnik: Resnik defined the similarity between two synsets to be the information content of their lowest common subsumer [23].

$$DistSim_{\mathrm{res}}(c_1,c_2) = IC\left(LCS(c_1,c_2)\right), \quad (6)$$

where IC is defined as

$$IC(c) = -\log p(c), \quad (7)$$

and $p(c)$ is the probability of a random word being an instance of concept c in a large corpus.

Jiang and Conrath: This similarity considers the information content of the lowest common subsumer and the two compared concepts [24].

$$\begin{aligned} &DistSim_{jcn}(c_1,c_2) \\ &= 1/\left[IC(c_1) + IC(c_2) - \left(2 \times IC\left(LCS(c_1,c_2)\right)\right)\right]. \end{aligned} \quad (8)$$

Lin: Lin builds on Resnik's measure of similarity, and adds a normalization factor consisting of the information content of the two concepts [25].

$$DistSim_{lin}(c_1,c_2) = \frac{2 \times IC\left(LCS(c_1,c_2)\right)}{IC(c_1) + IC(c_2)}. \quad (9)$$

Proposed similarity: We consider the length between each concept and the lowest common subsume.

$$\begin{aligned} &DistSim_{\mathrm{etri}}(c_1,c_2) \\ &= \left[1/\left(Depth(c_1) - Depth\left(LCS(c_1,c_2)\right) + 2\right)\right] \\ &\quad + \left[1/\left(Depth(c_2) - Depth\left(LCS(c_1,c_2)\right) + 2\right)\right]. \end{aligned} \quad (10)$$

4. Validation of SAT Based on Distance Similarity between Two ATs

The LAT is an important clue for restricting the SAT. In Q5, let us suppose that "behavior" and "person" are detected as the LATs, and PS_NAME (the SAT class related to a person) and OGG_BUSINESS (the SAT class related to a financial organization) are classified as the SATs. PS_NAME has a semantically close relationship to "person," whereas OGG_ BUSINEE is comparatively unrelated to "behavior" and "person." Thus, we reflected the relationship between the LAT and the SAT in the weight to express a degree of confidence.

Q5: A behavior disclosing any piece of information regarding any part of a given medium that a potential consumer would not want to know beforehand or a person doing this action. (Answer: spoiler)

5. Type Coercion Using LAT and SAT

Type coercion determines whether the answer candidate satisfies the AT requirement of the question. The relatedness score of the type coercion is our proposed similarity based on the taxonomy. We used three different type coercion methods for scoring the answer candidates.

A. Type Coercion Using LAT on Taxonomy

Wikipedia titles are connected with the taxonomy. Most questions have Wikipedia titles as answers, and thus it is significant that answer candidates, which are Wikipedia titles, be coerced by the LAT. For example, "Medici bank" is an answer candidate having high confidence for Q4 because "bank" (the LAT in Q4) and "Medici bank" are connected by the instance-of relation, as shown in Fig. 6.

B. Type Coercion Using the SAT and NE Class of Answer Candidates

Watson uses named-entity detection for type coercion [8]. This type coercion is based on the similarity between the NE classes of the LAT word and answer candidates. However, the NE class of the LAT word in the question cannot clearly reflect the semantic type of the expected answer. In addition, for Watson, questions without an LAT are not the subject of this type coercion.

As we mentioned earlier, we developed the SAT classifier to recognize what the question is asking. Our SAT classifier uses global features, and thus our classifier has a broader coverage. To label answer candidates with NE classes, we employed the NE recognizer of [10].

C. Type Coercion Using SAT on Taxonomy

We can coerce answer candidates into satisfying the SATs owing to a taxonomy with a connection to the SATs. For example, OGG_ECONOMY, which is the SAT of Q4, is linked to the "financial institution" node, as shown in Fig. 5. In addition, "Medici bank" is linked to "bank2" by an instance-of relation, as shown in Fig. 6. Thus, the similarity between OGG_ECONOMY and "Medici bank" is high.

IV. Experiment

1. Experimental Data and Environment

We constructed the experimental data for the LAT detector and SAT classifier based on ML. We collected about 86,400 questions from Naver (the largest portal site in Rep. of Korea) and *JangHak Quiz*, and manually annotated questions using an LAT and SAT. We chose 2,852 *Janghak Quiz* questions for the evaluation. Questions for the evaluation were also annotated with the answer. The rest of the questions were used as training data for machine learning. We conducted the experiment on a computer running Unbuntu Linux and equipped with an Intel Xeon CPU 3.50 GHz and 10 GB memory.

2. Question Analysis Experiment

We conducted three experiments. The purpose of the first experiment was to determine the effect of several features on the performance of the ML model. We adopted the structural support vector machine (sSVM) of [26] in the ML model. The second experiment was about improving the SAT performance using the taxonomic relatedness to LAT. The third experiment sought to evaluate the performance of the question analyzer using the evaluation data.

A. Evaluation Metrics

Our question analyzer was evaluated in terms of precision (P), recall (R), and $F1$-measure ($F1$), as shown in (11), (12), and (13), respectively.

$$P = \#\text{Correctly Detected } ATs / \#\text{Detected } ATs, \qquad (11)$$

$$R = \#\text{Correctly Detected } ATs / \# ATs \text{ in Evaluation Set}, \quad (12)$$

$$F1 = (2 \times P \times R) / (P + R). \qquad (13)$$

B. Effects of Features on the ML Model Performance

We employed the ML approach for LAT detection and SAT classification, and extracted various features from the questions. We carried out the experiments to judge whether each feature exerts a positive influence on the performance, as shown in Tables 4 and 5. We applied a ten-fold cross validation method to evaluate our ML-based models.

In the evaluation of the LAT detector, morpheme and W2V features have low precision and high recall, whereas a parsing feature has very high precision and extremely low recall. We analyzed the degree of contribution of the parsing, SRL, and W2V features from the baseline system. The baseline system comprises position, morpheme, NE, chunk, QT, and QF

Table 4. Results of evaluation regarding the influence of features on the performance of the ML-based LAT detector.

Features	Precision (%)	Recall (%)	*F*1 (%)
Morpheme	46.94	83.70	60.14
Parsing	89.54	3.15	6.09
SRL	37.43	39.50	38.41
W2V	20.93	66.45	31.82
Baseline	55.28	81.82	65.81
Baseline + Parsing	86.86 (+31.58)	60.42 (−21.40)	71.26 (+5.45)
Baseline + SRL	57.86 (+2.58)	80.40 (−1.42)	67.29 (+1.48)
Baseline + W2V	52.40 (−2.88)	84.00 (+2.18)	64.53 (−1.28)
ALL	85.74 (+30.46)	70.07 (−11.75)	77.11 (+11.30)

(): difference from the baseline.

Table 5. Influence of features on the performance of an ML-based SAT classifier.

Features	Precision (%)	Recall (%)	*F*1 (%)
Morpheme	42.49	42.28	42.44
LAT	40.72	40.62	40.67
Parsing	14.34	14.30	14.32
SRL	13.20	13.17	13.19
W2V	33.81	33.72	33.76
Concept	31.21	31.13	31.17
Baseline	62.56	62.40	62.48
Baseline + LAT	71.09 (+8.53)	70.91 (+8.51)	71.00 (+8.52)
Baseline + Parsing	62.67 (+0.11)	62.51 (+0.11)	62.59 (+0.11)
Baseline + SRL	62.81 (+0.25)	62.64 (+0.24)	62.72 (+0.24)
Baseline + W2V	71.99 (+9.43)	71.80 (+9.40)	71.89 (+9.41)
Baseline + Concept	68.70 (+6.14)	68.52 (+6.12)	68.61 (+6.13)
ALL	80.55 (+17.99)	80.34 (+17.94)	80.45 (+17.97)

(): difference from the baseline.

features. Syntactic features (parsing and SRL) have a good effect on precision, whereas a conceptualization feature (W2V) exerts a beneficial influence on recall. Using all these features, the precision and *F*1 were increased by 30.46% and 11.3%, respectively.

We analyzed the performance of SAT classification using various combinations of features. Lexical features (morpheme and LAT) showed a better performance than syntactic features (parsing and SRL) and conceptual features (W2V and concept). For the baseline system, we used feature combinations of morpheme, NE, noun phrase, chunk, QT, and QF. LAT and conceptual features contribute to performance. Using all these features, the precision, recall, and *F*1 were 80.55%, 80.34%, and 80.45%, respectively.

Table 6. Evaluation of SAT validation method based on relatedness between two ATs.

Features	Precision (%)	Recall (%)	*F*1 (%)
Baseline (ALL – LAT)	76.86	76.67	76.77
Baseline + LAT feature	80.55 (+3.69)	80.34 (+3.67)	80.45 (+3.68)
Baseline + Hirst and St-Onge	77.63 (+0.77)	77.43 (+0.76)	77.53 (+0.76)
Baseline + Leacock and Chodorow	77.75 (+0.89)	77.55 (+0.88)	77.65 (+0.88)
Baseline + Wu and Palmer	77.56 (+0.70)	77.36 (+0.69)	77.46 (+0.69)
Baseline + Resnik	76.88 (+0.02)	76.68 (+0.01)	76.78 (+0.01)
Baseline + Jiang and Conrath	75.98 (−0.88)	75.79 (−0.88)	75.89 (−0.88)
Baseline + Lin	77.50 (+0.64)	77.30 (+0.63)	77.40 (+0.63)
Baseline + Proposed Sim.	79.55 (+2.69)	79.34 (+2.67)	79.45 (+2.68)
Baseline + LAT feature + Proposed Sim.	81.21 (+4.35)	81.00 (+4.33)	81.11 (+4.34)

(): difference from the baseline.

C. Evaluation of SAT Validation Based on Distance Similarity on Taxonomy

To evaluate the SAT validation method based on the taxonomic relatedness between LAT and SAT, we organized a baseline model, which is an ML model using the features other than the LAT feature. The LAT feature model is an ML model using all features included the LAT feature. Other models consist of a baseline model and SAT validation method based on the following distance similarity. The "Baseline + LAT feature + Proposed Sim" model is applied to the SAT validation based on our proposed similarity with the ML method using all features. A score of the distance similarity is used for a re-ranking of the SAT classification. An evaluation was conducted with the same configuration as in the previous subsection.

Table 6 shows that the ML method using the LAT feature has a better performance than similarity-based methods. With the exception of Jiang and Conrath, the SAT validation model based on the other similarity methods contributed to performance improvement. In particular, our proposed similarity showed the best performance among the similarity-based methods. The "Baseline + LAT feature + Proposed Sim" model shows that this combination creates a synergistic effect. Similarity algorithms based on information content did not significantly influence the improvement in SAT performance compared with length-based similarity algorithms.

D. Question Analyzer Performance Evaluation

Table 7 shows the results of the evaluation of the question analyzer. We experimented with the evaluation data of 2,852 questions for all models and functions. The precision of the SAT fell 7% from the ML-based approach using training data because questions in the training data were collected from Naver and *JangHak Quiz* at different rates (a ratio of about 8:2). In addition, questions collected from Naver had less complexity than those from *JangHak Quiz*. A high rate of Naver questions led to a high precision for the ML-based approach, but the evaluation data consisted of only *JangHak Quiz* questions.

Table 7 also shows the comparison results of the LAT performance between Watson and *WiseQA*. The rule model of LAT is the result achieved using only a rule-based approach such as with IBM's Watson system. *WiseQA* is the evaluation result of the hybrid approach. Compared with a rule-based approach, *WiseQA* achieved an improved recall of 11.04% without a large decline in precision. Because our proposed system is for a Korean QA, we cannot directly compare our system with an English QA system. However, we know that an ML approach based on a sequence-labelling algorithm is a great help in improving the recall of LAT detection.

Table 7. Question analyzer evaluation.

Models	Functions	Precision (%)	Recall (%)	*F*1 (%)
Rule	QF	97.09	96.16	96.63
	LAT (like Watson)	88.09	67.62	76.51
	SAT	31.12	30.74	30.93
ML	LAT	90.23	66.12	76.32
	SAT	72.54	71.65	72.10
Hybrid	LAT (WiseQA)	84.68	78.66	81.56
	SAT	73.55	72.64	73.09

Table 8. Impact of type coercion on *WiseQA*.

Methods	Precision @100% (%)
No-TyCor.	78.0
+WWN (like Watson)	81.0 (+3.0)
+SAT_NE	82.3 (+4.3)
+SAT_WWN	81.3 (+3.3)
All-TyCor (WiseQA)	83.6 (+5.6)

(): difference from No-TyCor

3. Type Coercion Experiment

We evaluated the impact of the three type coercion methods shown in Table 8. Without using a type coercion method, the QA system, which uses a syntactic similarity between the question and the sentence containing the answer candidates, is No-TyCor. WWN is a system using the relatedness between the LAT and the answer candidates based on taxonomy, such as IBM's Watson system. SAT_NE is a system using the relatedness between the SAT and NE class of the answer candidates on the NE class hierarchy. SAT_WWN is a system using the relatedness between the SAT and the answer candidates on the taxonomy. All-TyCor is the system to which the three type coercions were applied.

Table 8 shows the impact of our individual type coercion method. Our system without type coercion answered only 78.0% of the questions correctly. The system with SAT_NE was the most effective, whereas the precision of the system with WWN was less effective. The system with all type coercion methods achieved a precision of 83.6%.

Figure 7 shows the impact of type coercion for different percentages of questions answered by our system. The horizontal axis of the graph shows the percentage of questions answered and the left vertical axis indicates the percentage of those questions that were correctly answered. Here, Y%@X% indicates the precision of Y% in X% of the questions answered; for X% of the questions for which the system was most confident in its answer, it answered Y% of those questions correctly [8]. The right vertical axis indicates the difference of precision between All-TyCor and WWN on the interval between the percentages of the questions answered. The difference in precision between @75% and @100% is significant. This result shows that SATs have a substantial effect on the performance when a system with WWN has a low confidence in answering.

V. Conclusion

Our hybrid approach to LAT detection achieved a higher recall rate than a rule-based approach. Our ML-based approach is based on sequence labeling using sSVM. Compared with Watson's rule-based approach, our LAT detector achieved a recall improvement of 11.04% without a large decline in precision.

We proposed an SAT classifier and a validation method based on the taxonomic relatedness using various similarity measures. Our proposed similarity helped improve the performance of the SAT classifier, which achieved a precision rate of 73.55%.

For type coercion using both an LAT and SAT, we constructed an answer-type taxonomy with a semantic relationship between the two ATs. Our taxonomy consists of linking UWordMap and KorLex (so-called "*WiseWordNet*"), *WiseWordNet* and NE classes, and *WiseWordNet* and Wikipedia titles.

We proposed three strategies for type coercion using an LAT and SAT based on the taxonomy. We found that the combination of individual type coercion created a synergistic effect..

WiseQA is a Korean question and answering system.

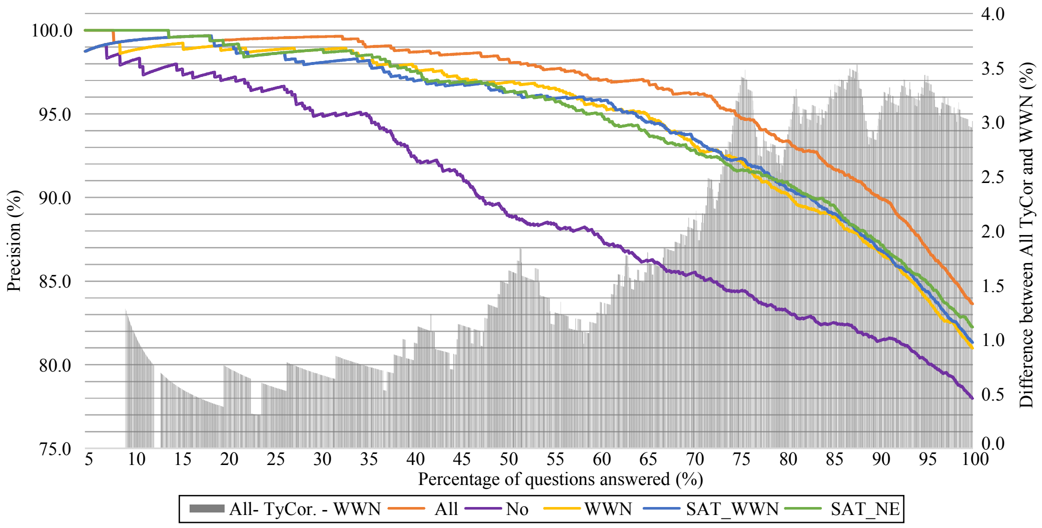

Fig. 7. Impact of type coercion on the percentage of questions answered (2,852 JangHak quiz questions).

Nevertheless, our approach can be applied for different languages and evaluation sets. Compared with IBM's Watson system, the superiority of our system is as follows. First, ML-based LAT detection contributed to an improvement in recall. Second, an SAT classifier and validation method using taxonomic relatedness were developed for type coercion related to the SAT. Third, we constructed an answer-type taxonomy with a semantic relationship between the LAT and SAT. Finally, we proposed a type coercion approach using the both ATs on the taxonomy. We found that type coercion using an SAT had a positive effect on the QA system.

References

[1] J. Burger et al., "Issues, Tasks and Program Structures to Roadmap Research in Question & Answering," *Document Understanding Conf. Roadmapping Documents*, 2001, pp. 1–35.

[2] J. Chu-Carroll et al., "IBM's PIQUANT II in TREC 2004," *Proc. TREC*, Gaithersburg, MD, USA, Nov. 16–19, 2004, pp. 184–191.

[3] D. Moldovan et al., "The Structure and Performance of an Open-Domain Question Answering System," *Proc. of Annu. Meeting ACL*, Hong Kong, China, Oct. 3–6, 2000, pp. 563–570.

[4] P.M. Ryu, M.G. Jang, and H.K. Kim, "Open Domain Question Answering Using Wikipedia-Based Knowledge Model," *Inform. Process. Manage.*, vol. 50, no. 5, Sept. 2014, pp. 683–692.

[5] P.C. Chen, M.J. Zhuang, and C.J. Lin, "Using Wikipedia and Semantic Resources to Find Answer Types and Appropriate Answer Candidates Sets in Question Answering," *Open Knowl. Base Question Answering Workshop COLING*, Osaka, Japan, Dec. 2016.

[6] D.A. Ferrucci, "Introduction to 'This is Watson'," *IBM J. Res. Develop.*, vol. 56, no. 3.4, May–June 2012, pp. 1:1–1:15.

[7] A. Lally et al., "Question Analysis: How Watson Reads a Clue," *IBM J. Res. Develop.*, vol. 56, no. 3.4, 2012, pp. 2:1–2:14.

[8] J.W. Murdock et al., "Typing Candidate Answers Using Type Coercion," *IBM J. Res. Develop.*, vol. 56, no. 3.4, May–June 2012, pp. 7:1–7:13.

[9] M.A. Pasca and S.M. Harabagiu, "High Performance Question/Answering," *Proc. Annu. Int. ACM SIGIR*, New Orleans, LA, USA, Sept. 2001, pp. 366–374.

[10] C.K. Lee et al., "Fine-Grained Named Entity Recognition Using Conditional Random Fields for Question Answering," *Proc. Asia Conf. Inform. Retrieval Technol.*, Singapore, Oct. 16–18, 2006, pp. 581–587.

[11] S.J. Lim et al., "Domain-Adaptation Technique for Semantic Role Labeling with Structural Learning," *ETRI J.*, vol. 36, no. 3, June 2014, pp. 429–438.

[12] C. Park et al., "Korean Coreference Resolution with Guided Mention Pair Model Using the Deep Learning," *ETRI J.*, vol. 38, no. 6, Dec. 2016, pp. 1207–1217.

[13] T. Mikolov et al., "Distributed Representations of Words and Phrases and Their Compositionality," *Proc. Int. Conf. Neural Inform. Process. Syst.*, Lake Tahoe, NV, USA, Dec. 5–10, 2013, pp. 3111–3119.

[14] D.W. Zhang et al., "Chinese Comments Sentiment Classification Based on Word2vec and SVM^{perf}," *Expert Syst. Applicat.*, vol. 42, no. 4, Mar. 2015, pp. 1857–1863.

[15] A. Toral et al., "A Study on Linking Wikipedia Categories to Wordnet Synsets Using Text Similarity," *Proc. Recent Adv. Natural Language Process.*, 2009, pp. 449–454.

[16] S. Fernando and M. Stevenson, "Mapping WordNet Synsets to Wikipedia Articles," *LREC Conf*, Turkey, May 2012, pp. 590–596.

[17] H.S. Choe, *Construction and Application of Large-Scale Korean User-Word Intelligent Network*, Ph.D. dissertation, University of Ulsan, Rep. of Korea, 2007.

[18] A.S. Yoon et al., "Construction of Korean WordNet 'KorLex

https://doi.org/10.4218/etrij.17.2816.0066

1.5’,” *J. KIISE: Softw. Applicat.*, vol. 36, no. 1, 2009, pp. 95–126.

[19] J. Chu-Carroll et al., “Textual Resource Acquisition and Engineering,” *IBM J. Res. Develop.*, vol. 56, no. 3.4, May–June 2012, pp. 4:1–4:11.

[20] G. Hirst and D. St-Onge, “Lexical Chains as Representations of Context for the Detection and Correction of Malapropisms,” *WordNet: An Electronic Lexical Database*, Cambridge, MA, USA: MIT Press, 1998, pp. 305–332.

[21] C. Leacock and M. Chodorow, “Combining Local Context and WordNet Similarity for Word Sense Identification,” *WordNet: An Electronic Lexical Database*, Cambridge, MA, USA: MIT Press, 1998, pp. 265–283.

[22] Z. Wu and M. Palmer, “Verbs Semantics and Lexical Selection,” *Proc. Annu. Meeting ACL*, Las Cruces, New Mexico, June 27–30, 1994, pp. 133–138.

[23] P. Resnik, “Using Information Content to Evaluate Semantic Similarity,” *Proc. Int. Joint Conf. Artificial Intell.*, Montreal, Canada, Aug. 20–25, 1995, pp. 448–453.

[24] J.J. Jiang and D.W. Conrath, “Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy,” *Proc. Conf. Res. Comput. Linguistics*, Taiwan, 1997.

[25] D. Lin, “An Information-Theoretic Definition of Similarity,” *Proc. Int. Conf. Mach. Learning*, July 24–27, 1998, pp. 296–304.

[26] C.K. Lee and M.G. Jang, “A Prior Model of Structural SVMs for Domain Adaptation,” *ETRI J.*, vol. 33, no. 5, 2011, pp. 712–719.

Jeong Heo received his BS and MS degrees in computer science from the University of Ulsan, Rep. of Korea, in 1999 and 2001, respectively. He received his PhD from the Department of Electrical/Electronic and Computer Engineering from University of Ulsan in 2017. He is a researcher of the Language Intelligence Research Group at the ETRI, Daejeon, Rep. of Korea. His research interests include big data analysis, natural language processing, information retrieval, and question answering.

Hyung-Jik Lee received his BS and MS degrees in electronic engineering from Kyungpook National University, Daegu, Rep. of Korea, in 1998 and 2000, respectively. Since 2000, he has been a principal researcher at the ETRI, Daejeon, Rep. of Korea. His research interests include big data analysis, natural language processing, machine learning, and question answering.

Ji-Hyun Wang received his BS and MS degrees in computer engineering from Chonbuk, National University, Cheongju, Rep. of Korea, in 1996 and 1998, respectively. Currently, he is a senior researcher at the ETRI, Daejeon, Rep. of Korea. His research interests include natural language processing, machine learning, question answering, and artificial intelligence.

Yong-Jin Bae received his BS degree in computer education from Mokwon University, Daejeon, Rep. of Korea, in 2012 and the MS degree in computer software and engineering from the UST, Daejeon, Rep. of Korea. Currently, he is a researcher at the ETRI, Daejeon, Rep. of Korea. His research interests include social big data analytics, knowledge engineering, and question answering.

Hyun-Ki Kim is a researcher in the language intelligence research group at ETRI, Daejeon, Rep. of Korea. He received his BS and the MS degrees in computer science from Chonbuk National University, Cheongju, Rep. of Korea, in 1994 and 1996, respectively. He received his PhD in computer engineering from the University of Florida at Gainesville, USA in 2005. His research interests include natural language processing, machine learning, question answering, and social big data analytics.

Cheol-Young Ock is a professor of the School of IT Convergence, University of Ulsan, Rep. of Korea. He received his BS (1982), MS (1984), and PhD (1993) degrees in computer engineering from the National University of Seoul, Rep. of Korea. He has been a visiting professor at the Russia Tomsk Institute, Russia (1994), and Glasgow University, UK (1996). He was also a chairman of sigHCLT (2007 to 2008) in KIISE, Rep. of Korea. He has been a visiting researcher at the National Institute of Korean Language, Rep. of Korea (2008). He received an honorary doctorate from the School of IT, National University of Mongolia, (2007), and earned a medal for Korean development from the Korean government (2016). He has been constructing a Korean WordNet, namely Ulsan Word Map (UWordMap) since 2002. His research interests include natural language processing (WSD), machine learning, and text mining.