ISSN:0975-9646

Semantic Answer Validation using Universal Networking Language

Partha Pakray, Utsab Barman, Sivaji Bandyopadhyay, Alexander Gelbukh

Computer Science & Engineering Dept.,
Jadavpur University,Kolkata, India.

***Abstract*— we present a rule-based answer validation (AV) system based on textual entailment (TE) recognition mechanism that uses semantic features expressed in the Universal Networking Language (UNL). We consider the question as the TE hypothesis (H) and the supporting text as TE text (T). Our proposed TE system compares the UNL relations in both T and H in order to identify the entailment relation as either validated or rejected. For training and evaluation, we used the AVE 2008 development set. We obtained 58% precision and 22% F-score for the decision "validated."**

***Keywords*— question answering, textual entailment, Universal Networking Language, AVE 2008 data sets.**

I. Introduction

A huge amount of information available nowadays in machine-readable form cannot be managed without efficient means of automatic search. A traditional approach to search is information retrieval (IR), a process that takes keywords describing the user's information need, such as "Bangkok location" and present the user with a set of documents (which can be a very large set) that contain these keywords, leaving to the user their manual analysis that hopefully would eventually lead to the satisfaction of his or her information need, which itself, the need, remains unknown to the system.

Question answering (QA) systems bring the task of the search to a radically new level, allowing the user to directly formulate his or her information need in the form of a natural language question, such as "Where is Bangkok located?". As output, an QA system gives a direct answer to the question: "In Thailand," eliminating the need for the user to manually analyze hundreds of documents in order to find the desired answer.

Internally, basing on the analysis of the available documents, the system may generate several possible answers, e.g., "In Asia"; "In Thailand"; "on the banks of Chao Phraya river"; "from 950 dollars". The system internally knows which snippet of text has led it to which answer; such snippets are called supporting text, e.g.: "We offer flights for such locations as Bangkok from 950 dollars". Some of these answers can be plainly incorrect, i.e., not really implied by the corresponding snippet, while some of the correct answers can be more desirable than others. Only one answer is supposed to be presented to the user and the final decision of the system.

This paper is devoted to the mechanism of choice between such alternative answers generated by a QA system. Namely, we present an *answer validation* (AV) system based on *recognizing textual entailment* (RTE) task that uses the *Universal Networking Language* (UNL) semantic representation. Let us briefly introduce these three concepts.

Answer validation. The Answer Validation Exercise (AVE) is a task recently introduced in the QA@CLEF competition. The AVE task is aimed at developing systems for automatic evaluation of the answers produced by question answering systems. One of the aims in the research on AV systems is to identify the factors useful for improvement of QA systems. There have been three AVE competitions so far: AVE 2006 [1], AVE 2007 [2] and AVE 2008 [3].

An AV system receives a set of questions, each one of that supplied with a set of possible answers with the supporting text for each answer—which models the output of an imaginary QA system. For each answer, the AV system it should return one of the following three possible judgments:

- VALIDATED, if the AV system considers the answer correct with respect to the supporting text. There is no restriction on the number of VALIDATED answers to the same question if several answers were presented to the system in input.
- SELECTED, if the answer is VALIDATED and in addition the system considers that this answer is the one that should be chosen as the output of a hypothetical QA system that the AV system is evaluating. Exactly one of the VALIDATED answers must be marked as SELECTED.
- REJECTED, if the AV system considers the answer incorrect or sees not enough evidence of its correctness. There is no restriction on the number of REJECTED answers.

The evaluation methodology has been improved over the years. In 2007, the AVE systems were asked to select only one VALID answer for every question from a set of possible answers. In 2006, several VALID answers for the same question were permitted. In 2008, the organizers increased the complexity of the data set by allowing all the answers to a given question to be incorrect. The task of the participating systems was to ensure that all the answers to such questions are marked as REJECTED.

Recognizing textual entailment. Recognizing textual entailment (RTE) is one of the recent challenges of natural language processing (NLP). Textual entailment (TE) is defined as a directional relationship between pairs of textual expressions. One expression in the pair is called the hypothesis (H) and the other text (T), The directional relation holds between H and T if the meaning of H can be inferred from the meaning of T as would typically be interpreted by people.

Textual Entailment has applications in many NLP tasks. For example, in summarization (SUM), the generated summary should be entailed by the input text. Paraphrases (PP) can be seen as mutual (bidirectional) entailment between T and H. In information extraction (IE), the extracted information should be entailed by the input text. In question answering (QA), the answer generated for a question after the information retrieval (IR) process must be entailed by the supporting snippet of the text.

Three Recognizing Textual Entailment (RTE) competitions—RTE-1 in 2005 [4], RTE-2 in 2006 [5] and RTE-3 in 2007 [6]—have been organized by the Pattern Analysis, Statistical Modeling and Computational Learning (PASCAL), a European Commission's IST-funded Network of Excellence for Multimodal Interfaces. In 2008, the fourth edition (RTE-4) [7] of the challenge was organized by National Institute of Standards and Technology (NIST) at the Text Analysis Conference (TAC).

At every new RTE competition, several new features were introduced. The TAC RTE-5 [8] challenge in 2009 included a separate search pilot task along with the main task. The TAC RTE-6 challenge [9] in 2010 included the main task and the novelty detection task along with the RTE-6 knowledge base population (KBP) validation pilot task. The RTE-6 did not include the traditional RTE main task, which was carried out in the first five RTE challenges—i.e. there have been no task to make entailment judgments over isolated T–H pairs drawn from multiple applications. In 2010, Parser Training and Evaluation using Textual Entailment challenge [10] was organized at SemEval-2.

The present work is a result of a continuation of our RTE system that has participated in TAC RTE-5 in 2009, Parser Training and Evaluation using Textual Entailment at SemEval-2 and in TAC RTE-2010.

Universal Networking Language. Universal Networking Language (UNL) [14, 15] is an artificial language designed to express information or knowledge in the form of semantic network with hyper-nodes. It has applications in the domains of machine translation (MT), information retrieval (IR) and multilingual document generation, to name just a few.

UNL defines a set of so-called universal words (UW), relations and attributes. UWs model concepts. The binary relationships among the (universal) words in a natural language sentence are specified as UNL relations. Attributes are properties of the UWs.

A semantic network expressed in UNL includes a set of binary relations; each binary relation relates the two UWs that hold the relation. A binary relation of UNL is expressed in the following format: <Relation>(<UW$_1$>, <UW$_2$>).

The paper is organized as follows. Related works are described in Section II. Section III describes the statistics of the corpus used in our experiments. Section IV presents our answer validation system. The experiments carried out on the development and test data sets are described in Section V along with the results. The conclusions are drawn in Section VI.

II. Related Works

In various AVE challenges, several methods have been applied. Most of these systems use some sort of lexical matching. A number of systems represent the texts as parse trees (e.g., syntactic or dependency trees) before accomplishing the actual task. Some of those systems use semantic relations (e.g., logical inference or semantic role labeling) for solving the text-and-hypothesis entailment problem.

The system [11] casts the AVE task as a recognizing textual entailment (RTE) problem and uses an existing RTE system to validate the answers. Additional information from named-entity (NE) recognizer, question analysis component, and some other modules was also considered in order to assist in making the final decision.

RAVE (Real-time Answer Validation Engine) [12] is a logic-based answer validator and selector designed for application in real-time question answering. RAVE uses the same tool chain for deep linguistic analysis and the same background knowledge as its predecessor (MAVE), which took part in the AVE 2007. However, a full logical answer check as in MAVE was not considered suitable for real-time answer validation since it requires parsing of all answer candidates. Therefore, RAVE uses a simplified validation model where the prover only checks whether the support passage contains a correct answer at all. This move from logic-based answer validation to logical validation of supporting snippets permits RAVE to avoid any parsing of answers, i.e., the system only uses a parse of the question and pre-computed snippet analyses. In this way very quick validation and selection can be achieved. Machine learning is used for assigning local validation scores using both logic-based and shallow features.

The system [13] uses a set of regular expressions in order to join the question and the answer into an affirmative sentence and afterwards applies several techniques of lexical–semantic inference in an attempt to detect whether the meaning of this sentence can be inferred by the meaning of the supporting text.

In this work, we use an RTE system in the way similar to [11], while we specially construct our RTE system for this task. We construct our hypothesis in the way similar to [13], and then use UNL as an underline semantic representation of the text and hypothesis for the entailment decisions.

III. Corpus Statistics

At the AVE 2008 challenge, separate sub-tasks for the following 11 languages were made available by the organizers: Basque, Bulgarian, German, English, Spanish, French, Italian, Dutch, Portuguese, Romanian, and Greek. We used the corpus for the English monolingual task.

The corpus is organized as a set of triplets <question, answer, supporting text>. The participating systems had to specify the organizers of the answer in terms of SELECTED, VALIDATED or REJECTED as described in

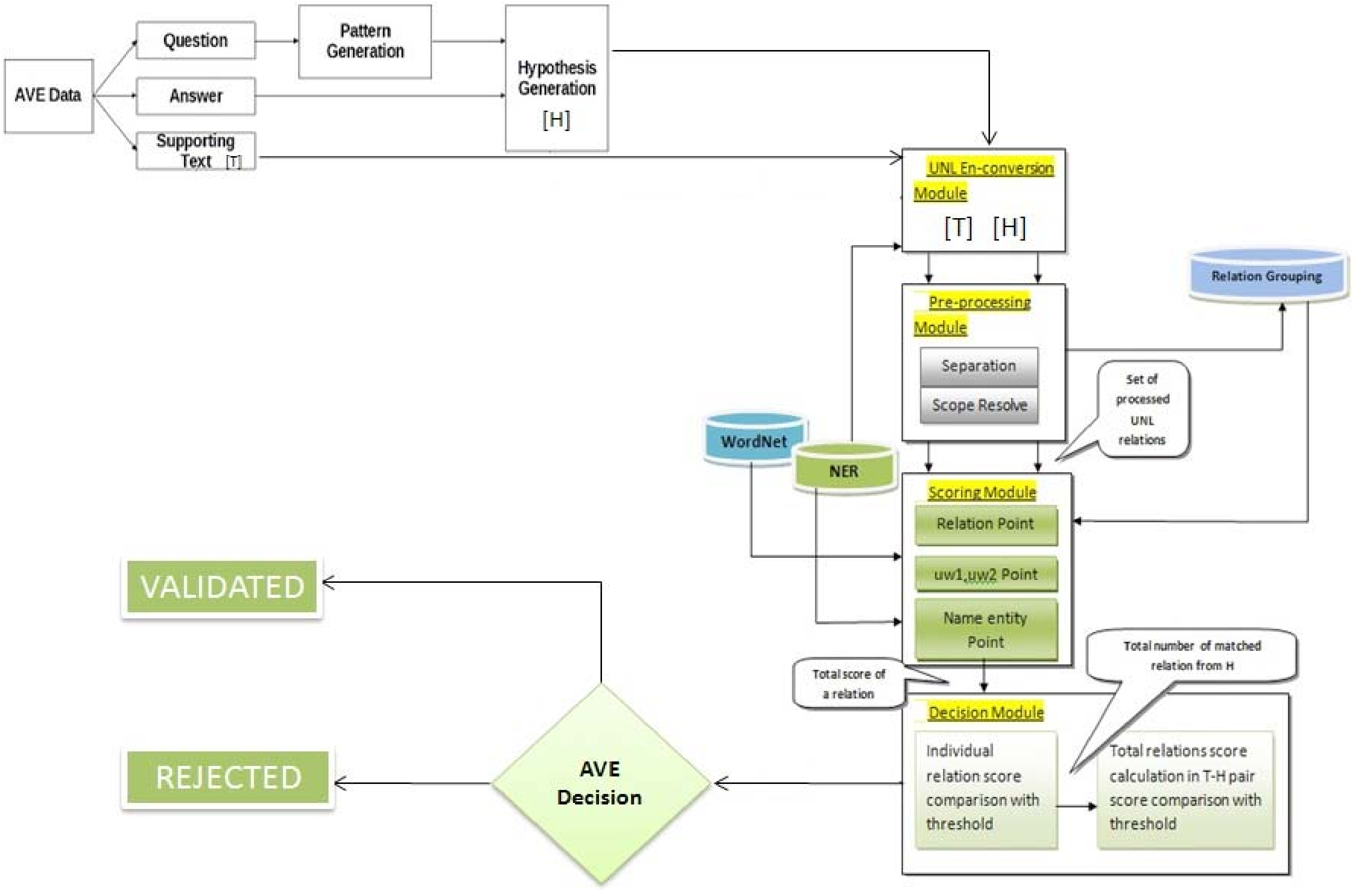

Fig. 3 System Architecture

Section I. The AVE 2008 development set's data format is shown in Figure 1.

<q id="83" lang="EN">
<q_str>Where is the Hermitage Museum?</q_str>
<a id="83_2" value="REJECTED">
<a_str>Birseck</a_str>
<t_str doc="en/p03/334819.xml">The Mesolithic period has some examples of portable art, like painted pebbles (Azilien) from Birseck, Eremitage in Switzerland, and in some areas, like the Spanish Levant, stylized rock art.</t_str>
</a>...</q>

Fig. 1 AVE 2008 Test Gold Data Format

In Figure 1, the data format *q_str* tag contains the question, *a* tags correspond to possible answers, *a_str* tag contains the answer itself and justification text is in the *t_str* tag.

The AVE 2008 data output format is shown in Figure 2. The output for a question–answer combination can be either VALIDATED, SELECTED or REJECTED.

q_id a_id [SELECTED | VALIDATED | REJECTED] confidence

Fig. 2 AVE 2008 Data Output Format

For evaluation, the SELECTED answers were compared against the QA systems of the main track.

IV. System Architecture

The architecture of our Answer Validation (AV) system is presented in Figure 3. The main components of our AV system are:

- the pattern generation module,
- the UNL en-conversion module,
- the pre-processing module,
- the scoring module, and
- the decision module.

In the sequel we describe each one of these modules.

A. Pattern Generation Module

First, we convert each question into an affirmative sentence that denotes the answer pattern and place the [answer] template in place of the appropriate answer. The pattern generation module is rule based. For example, for the question id 0061 (AVE-2008 test set) we obtain:

Question: Where was Joseph Fourier born?
Template: Joseph Fourier was born in [answer].

B. Hypothesis Generation Module

After pattern generation the [answer] template is replaced by the answer string forming the generated hypothesis.

Now, we have the text (T), which is the supporting text, and the hypothesis (H), which is the generated hypothesis. For example, for the same question id 0061 of the AVE-2008 test set, we generate the following hypotheses for each of the alternative answers:

H0061_1: Joseph Fourier was born in Paris.
H0061_2: Joseph Fourier was born in France.

C. UNL En-Conversion Module

The obtained T–H pairs are converted into UNL expressions using the UNL en-Converter. An example of a UNL expression of a generated hypothesis of AVE development data is shown in Figure 4.

```
[S:00]
{org:en}
The occupation of Kiri Te Kanawa is - opera singer
{/org}
{unl}
aoj(singer(icl>musician>thing).@entry.@present,occupation(icl
>acquiring>thing).@def)
obj(occupation(icl>acquiring>thing).@def,kiri(icl>surname,iof
>person))
nam(kiri(icl>surname,iof>person),te)
nam(te,kanawa)
mod(singer(icl>musician>thing).@entry.@present,opera(icl>cla
ssical_music>thing)){/unl}[/S]
```

Fig. 4. Generated hypothesis of AVE development data, in UNL

D. Pre-processing Module

Pre-processing consists of three steps:

- separation,
- scope resolution, and
- relation grouping.

Separation. From the UNL graphs of T and H, individual UWs are extracted using regular expressions. The regular expressions that are used to extract the individual UWs are as follows:

For UW1: [#] + [-a-z0-9R:._-ʾ&=*'`~\"\\ + [\\s]] + [\\
(.,]
For UW2: [,] + [-a-z0-9R:._- &=*'`~\"\\ + [\\s]] +
[\\(.#]

The relation name, scope id, constraint list and attribute list are separated from the UNL relation graph. All relations are put up into a logical set in some specific format as per our system requirement:

[Relation Name] [Relation Scope ID]
{[UW1][UW1 Scope id], [UW2][UW2 Scope id]}

Scope Resolution. The specific task at this step is to resolve the scope id of UNL relations. Consider, for example, the UNL relation format shown in Figure 5. In the fourth relation Cob we find a scope id ':01' in the place of UW_2 that specifies the relation between the present UNL graph and the other UNL graph specifying UW_2. In the sentence the main subject / noun in focus is 'Pfizer' and the other noun 'children' in the predicate part has less focus. But the second noun is directly affected by the action of the first one that has occurred in parallel. The UNL specification defines the relation Cob precisely as a relation that "defines a thing that is directly affected by an implicit event done in parallel or an implicit state in parallel."

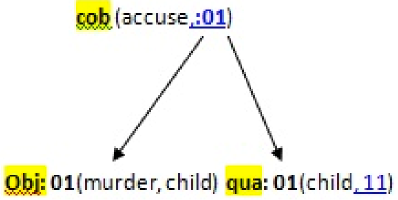

Fig. 5 UNL graph of Cob relation

Relation Grouping. Relation grouping process groups UNL relations that are semantically identical [16]. UNL relations form a semantic hyper-network, and UW_1 and UW_2 are two nodes of the graph that have a relation specified by the relation name in the UNL graph. Our strategy is based on the thematic roles of different relations. Table I shows different relation groups and the set of relations in each group.

TABLE I
UNL RELATION GROUPING

Group Name	Relations
Agent	*Agt, cag, aoj, cao, ptn*
Object	*Obj, cob, opl, ben*
Place	*Plc, plf, plt*
Instrument	*Ins, met*
State	*Src, gol, via*
Time	*Tim, tmf, tmt, dur*
Manner	*Man, bas*
Logical	*And, or*
Concept	*Equ, icl,iof*
Cause	*Con, pur, rsn*
Sequence	*Coo, seq, cnt, mod, nam, per, pof, pos, qua*

E. Scoring Module

The scoring module calculates the similarity score between two T–H UNL relations. The module assigns points to each relation pair using certain set of rules. The corresponding rules are as follows:

- Relation grouping rule
- UW Rule
- Name Entity Rule

Relation grouping rule. This rule checks whether the two UNL relations or UNL graphs, one from the text and another from the hypothesis, are in the same relation group. If so, it is considered as a match and one point is assigned to the relation pair.

UW rule. This rule checks whether UWs in the two matched UNL relations or graphs are same or belong to the same synset, i.e., have same meaning. We used the riWordNet[1] for synset matching. One point is considered as the score for each UW match. In case of the presence of scope id in the place of any UW, the comparison will be done from the UW list created at the scope resolution step.

Name Entity Rule. Assume *n* named entities in H and *m* Named Entities in T, and let *k* be the number of named entities that are present in both H and T. Then the number of points for named entity similarity will be calculated as the fraction of the named entities in the hypothesis that match, i.e., k / n. The composite score of a T–H pair is calculated as follows:

$$\text{Total Score (TS)} = \text{Relation Match Point (RMP)} + \text{UW}_1 \text{ point } (\text{UW}_1) + \text{UW}_2 \text{ point } (\text{UW}_2) + (k / n) \quad (1)$$

F. Decision Module

Our decision procedure consists of two steps:

– Individual Relation Pair decision and
– Total Relations Score Calculation.

First, the decisions are made individually by each pair. Then, the final decision is made by the total score.

Individual Relation Pair decision. The total score of individual relation pair is calculated using the equation (2) below. The maximum value of the total score (TS_{max}) for each individual relation pair is calculated as (4).

$$TS = RMP + UW_1 + UW_2 + (k / n) \quad (2)$$

$$TS_{max} = RMP_{max} + UW_{1\,max} + UW_{2\,max} + (k/n)_{max} \quad (3)$$

$$RMP_{max} = UW_{1\,max} = UW_{2\,max} = (k/n)_{max} = 1 \quad (4)$$

$$\text{Hence } TS_{max} = 4 \quad (5)$$

The minimum value of the total score (TS_{min}) for each individual relation pair has been observed as 3.5 from the training sets of the various RTEs. If the total score of a relation pair falls between 3.5 and 4, the relation pair is considered as a match.

Total Relations Score Calculation. If there are H_n UNL relations in H and T_n UNL relations in T, then the number of matched relation pairs (M_n) identified. The final score (FS) for the T–H pair is calculated as (M_n / H_n). It has been observed from the training sets of the various RTEs that the minimum value of FS is 0.96 for the T–H pair to be considered as entailment. Hence, if the FS score for a T–H pair is 0.96 or above, then the T–H pair is considered as entailment.

V. Experimental Results

Our Answer Validation system has been tested on the AVE 2008 development set for English. This set consists of 195 pairs, of which only 21 are positive (10.77% of the total number of pairs). The recall, precision and F-measure values on the development data obtained over correct VALIDATED answers are shown in Table II.

TABLE II
AVE 2008 DEVELOPMENT SET PRECISION, RECALL AND F-SCORE

AVE Development Set	Result
VALIDATED in the development set	21
VALIDATED in our system	69
VALIDATED match	17
Precision	0.80
Recall	0.24
F-score	0.37

The AVE 2008 English annotated test set consists of 1055 pairs and the number of correct VALIDATED answers is 79 (7.5% of the total). The recall, precision and F-measure values on the test data obtained over correct answers are shown in Table III.

TABLE III
AVE 2008 TEST SET PRECISION, RECALL AND F-SCORE.

AVE Test Set	**Result**
VALIDATED in the test set	79
VALIDATED in our system	339
VALIDATED match	46
Precision	0.58
Recall	0.13
F-score	0.22

VI. Conclusion

Our results show that a semantic based textual entailment approach appropriately tackles the answer validation (AV) problem.

Our experiments have been carried out for a semantic and syntactic based AV task. In our future work, we plan to carry out detailed error analysis of the present system and identify ways to overcome the errors.

References

[1] Peñas, A., Rodrigo, Á., Sama, V., Verdejo, F.: "Overview of the answer validation exercise 2006". Working Notes of CLEF 2006. (2006)

[2] Peñas, A., Rodrigo, Á, Verdejo, F.: "Overview of the Answer Validation Exercise 2007". Working Notes of CLEF 2007. (2007)

[3] Rodrigo, Á., Peñas, A., Verdejo, F.: "Overview of the answer validation exercise 2008". Working Notes of CLEF 2008. (2008)

[4] Dagan, I., Glickman, O., Magnini, B.: "The PASCAL Recognising Textual Entailment Challenge". Proceedings of the First PASCAL Recognizing Textual Entailment Workshop, 2005. (2005)

[5] Bar-Haim, R., Dagan, I., Dolan, B., Ferro, L., Giampiccolo, D., Magnini, B., Szpektor, I.: "The Second PASCAL Recognizing Textual Entailment Challenge". Proceedings of the Second PASCAL Challenges Workshop on Recognising Textual Entailment, Venice, Italy, 2006. (2006)

[6] Giampiccolo, D., Magnini, B., Dagan, I., Dolan, B.: "The Third PASCAL Recognizing Textual Entailment Challenge". In Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing, Prague, Czech Republic, 2007. (2007)

[7] Giampiccolo, D., Dang, H. T., Magnini, B., Dagan, I., Cabrio, E.: "The Fourth PASCAL Recognizing Textual Entailment Challenge". In TAC 2008 Proceedings.

[8] Bentivogli, L., Dagan, I., Dang. H.T., Giampiccolo, D., Magnini, B.: "The Fifth PASCAL Recognizing Textual Entailment Challenge". In TAC 2009 Workshop, National Institute of Standards and Technology Gaithersburg, Maryland USA, 2009.

[1] www.rednoise.org/rita/wordnet/

[9] Bentivogli, L., Clark, P., Dagan, I., Dang, H.T., Giampiccolo, D.: "The Sixth PASCAL Recognizing Textual Entailment Challenge". In TAC 2010 Workshop, National Institute of Standards and Technology Gaithersburg, Maryland USA, 2010.

[10] Yuret, D., Han, A., Turgut, Z.: "SemEval-2010 Task 12: Parser Evaluation using Textual Entailments". Proceedings of the SemEval-2010 Evaluation Exercises on Semantic Evaluation, 2010. (2010)

[11] Wang, R., Neumann, G.: "Information Synthesis for Answer Validation". In CLEF 2008Working Notes.

[12] Glöckner. I.: "University of Hagen at CLEF 2008: Answer Validation Exercise", In CLEF 2008 Working Notes.

[13] Ferrandez, O., Munoz, R., Palomar, M.: "A Lexical–Semantic Approach to AVE", In CLEF 2008 Working Notes.

[14] UNDL Foundation. Universal networking language (unl) specifications, edition 2006, August 2006. http://www.undl.org/unlsys/unl/unl2005-e2006

[15] Cardeñosa, J., Gelbukh, A., Tovar, E. (eds.) "Universal Networking Language: Advances in Theory and Applications". Research on Computing Science, vol. 12, IPN, Mexico, 2005, 443 pp. http://www. cicling.org/2005/UNL-book/

[16] Ishizuka, M.. "A Solid Foundation of Semantic Computing toward Web Intelligence", School of Information Science and Technology. http://www.jst.go.jp/sicp/ws2010_austria/presentation/presentation_14.pdf