Vol. 41 No. 2 Transactions of Information Processing Society of Japan Feb. 2000

Technical Note

An Extended SD-Form Semantics Model

Masahiro Wakiyama,[†1] Shouta Yoshihara,[†2] Hideki Noda,[†3]
Koichi Nozaki[†4] and Eiji Kawaguchi[†3]

The authors' semantics model, titled SD-Form Semantics Model, was proposed as a general scheme to tackle the quantitative semantic studies of natural language. However, the basic model was a little too simple to cover all aspects of meaning in language. In order to make the model more flexible, this article introduces a few new ideas. These ideas do not change the basic framework of the model in any way; rather, they extend it to a more generalized form.

1. Introduction

The authors recently proposed a semantics model that uses an SD-Form as a meaning description language[3]. In that model (the basic model) all the simple concepts (described by concept symbols) were introduced to be independent of other concepts. As a result, human knowledge about a concept and its property values was not well formalized in the model. Another problem was the lack of a scheme to facilitate "ambiguous reasoning."[1),2)]

In the present article, we will extend our basic model to a more generalized one by introducing new ideas.

In Section 2 we discuss the computation scheme of the elaboration score for a concept that has concept properties. Then in Section 3 an opposite concept pair is introduced to allow a larger semantic distance between them. In Section 4 we generalize the elaboration relation to a specialization relation to provide an ambiguous reasoning mechanism. Finally, we offer some brief concluding remarks.

2. A Knowledge-based Syntactic Elaboration

In English, "big house" and "young lady" are common expressions. So, we are not so much surprised at hearing such expressions. While, "honest house" and "turtle lady" are not very familiar to us, and may cause us an unusual feeling if we hear them. This is because we know that some houses are big and some ladies are young in reality, but we do not know if there is some "honest" house, or "turtle" lady. This kind of situation also should be reflected in our model. More specifically, the semantic difference score between "house" and "honest house" should be much larger than that between "house" and "big house."

We introduce two new connectors "*prof*" (property of) and "*vaof*" (value of) to define a knowledge-based syntactic elaboration. The new idea is an $\boldsymbol{ELAB_{synt}}$ relation having a score value as small as the $\boldsymbol{ELAB_{know}}$ relation. The following examples illustrate the usage of the new connectors. The data are to be provided as system knowledge.

⟨**Example 2-1**⟩
(A) (HUMAN)*prof*
([(SEX)*vaof*([MALE,FEMALE]),
(AGE-LEVEL)*vaof*
([YOUNG, ADULT, SENIOR])])
(HUMAN has properties of SEX and AGE-LEVEL with property values
[MALE,FEMALE] and
[YOUNG, ADULT, SENIOR]
respectively.)
(B) (HOUSE)*prof*
([(GRADE)*vaof*([FANCY,SHABBY]),
(SIZE)*vaof*([BIG, SMALL]),
(COLOR)*vaof*([WHITE, RED, BROWN])])
(HOUSE has properties of GRADE, SIZE, and COLOR with property values
[FANCY,SHABBY],
[BIG, SMALL]
[WHITE, RED,BROWN]
respectively.)

We have set up the new $\boldsymbol{ELAB_{synt}}$ scores

†1 Department of Control & Information Systems Engineering, Kitakyushu National College of Technology
†2 Department of British-American Culture Studies, Junshin Junior College
†3 Department of Electrical, Electronic & Computer Engineering, Kyushu Institute of Technology
†4 Information Science Center, Nagasaki University

in **SDENV-3** (the latest version of our experimental system) as follows. They use a Prolog-like description.

$$\boldsymbol{ELAB}_{\boldsymbol{synt}}(D_1, D_1/D_{2i}, n)$$
$$\text{:- } (D_1)prof([(D_{11})vaof(D_{21})], \ldots, (D_{1i})vaof(D_{2i})], \ldots), n \text{ is } 3.$$

Otherwise,

$$\boldsymbol{ELAB}_{\boldsymbol{synt}}(D_1, D_1/D_{2i}, n)$$
$$\text{:- } n \text{ is } \boldsymbol{SI}(D_{2i}) + 1.$$

Namely, if the system has "*prof*" and "*vaof*" type knowledge, the corresponding $\boldsymbol{ELAB}_{\boldsymbol{synt}}$ scores are reduced to smaller values (3 semit). If not, they are the same as before.

For **Example 2-1**, we have:

$$\boldsymbol{ELAB}_{\boldsymbol{synt}}(\text{HUMAN}, \text{HUMAN/YOUNG}) = 3,$$
$$\boldsymbol{ELAB}_{\boldsymbol{synt}}(\text{HOUSE}, \text{HOUSE/FANCY}) = 3,$$
$$\boldsymbol{ELAB}_{\boldsymbol{synt}}(\text{HOUSE}, \text{HOUSE/HONEST}) = 11.$$

This extension will make the SD-Form model more adaptable to real life than before. We have already implemented this extension in our system (**SDENV-3**).

We need a more detailed definition for $\boldsymbol{ELAB}_{\boldsymbol{synt}}(D_1, D_1/(\cdots)para(\cdots)para\cdots para(\cdots))$ type cases. However, this is beyond the scope of the present article.

3. Semantic Difference between Opposite Concepts

"Good" and "bad", or "big" and "small", are regarded as opposite concepts. In our basic model, however, we did not have any mechanism for treating them as opposite. We think the semantic difference between "good" and "bad" should be larger than that between "good" and "new", because "good" and "bad" are opposite, while "good" and "new" are not related. This requires a new framework in the model.

Let D'_1 and D'_2 be a pair of opposite concepts defined by a $(D'_1)oppo(D'_2)$ type knowledge, where "*oppo*" is a new connector in the model. We denote the modified algorithm for detecting the nearest common ancestor by

$$\boldsymbol{NCOA}^*(D'_1, D^*_0, D'_2, n_1, n^*_0, n_2).$$

The "D^*_0" in this expression does not represent any concrete concept. It is a symbol to designate the formal ***NCOA*** of D'_1 and D'_2, while D_0 is the ***NCOA*** detected by the old algorithm, namely,

$$\boldsymbol{NCOA}(D'_1, D_0, D'_2, n_1, n_0, n_2).$$

When many opposite concept pairs appear in one situation, we may describe them as $D^*_{01}, D^*_{02}, \ldots$, etc. If such opposite concept pairs are included in a nearest common ancestor detection process, we denote the overall ***NCOA*** by the symbol D^*_0 (c.f. **Fig. 1**).

The new semantic difference scores are defined by the following:

(1) If $(D'_1)oppo(D'_2)$ is registered as a piece of knowledge, then the semantic difference is:
$$\boldsymbol{DIFF}^*(D'_1, D'_2) = n^*_0 = 2n_0.$$

(2) If $(D'_1)oppo(D'_2)$ is not registered in the system, then
$$\boldsymbol{DIFF}^*(D'_1, D'_2) = n^*_0 = n_0.$$

Therefore, general $\boldsymbol{DIFF}^*(D_1, D_2)$ score calculation is executed by the following algorithm:

⟨$\boldsymbol{DIFF}^*(\boldsymbol{D}_1, \boldsymbol{D}_2)$ **algorithm**⟩

If $(D'_1)oppo(D'_2)$ is true in the system, then
$\boldsymbol{ELAB}(D'_1, D_1, m_1)$, $\boldsymbol{ELAB}(D'_2, D_2, m_2)$,
$\boldsymbol{NCOA}^*(D'_1, D^*_0, D'_2, n_1, n^*_0, n_2)$,
$n^*_1 = 2n_1 + m_1, n^*_2 = 2n_2 + m_2$,
$n^*_0 = n^*_1 + n^*_2$.

Otherwise,
$n^*_0 = n_0$
where, $\boldsymbol{NCOA}(D_1, D_0, D_2, n_1, n_0, n_2)$.

Let us consider the following examples:

⟨**Example 3-1**⟩

System Knowledge:

(MAN)*incl*([TOM, BOB]),
(MARRY)*oppo*(DIVORCE),
(FUTURE)*oppo*(PAST).

Statements:

D_1 = [*s*((TOM)*plus*(KATE)), *v*(MARRY/FUTURE)]
(Tom and Kate will marry.)

D_2 = [*s*((BOB)*plus*(KATE)), *v*(DIVORCE/PAST)]
(Bob and Kate divorced.)

NCOA*:

D_0 = [*s*((MAN/SOME)*plus*(KATE)), $v(D^*_{01}/D^*_{02})$].

In this case we can compute the ***DIFF**** score as follows:

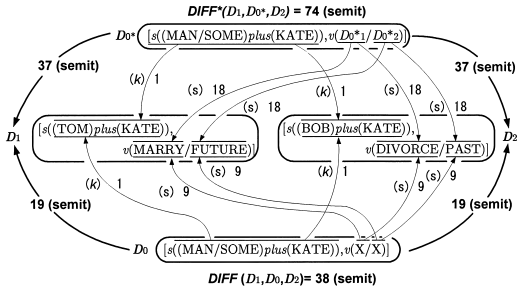

Fig. 1 Computation of $\boldsymbol{DIFF}^*$ scores.

$$\begin{aligned}&\boldsymbol{DIFF}^*(D_1, D_2)\\ &\quad = \boldsymbol{DIFF}(\text{TOM}, \text{BOB})\\ &\qquad +\boldsymbol{DIFF}^*(\text{MARRY}, \text{DIVORCE})\\ &\qquad +\boldsymbol{DIFF}^*(\text{FUTURE}, \text{PAST})\\ &\quad = 2 + 36 + 36 = 74 \text{ (semit.)}\end{aligned}$$

In this example D^*_{01} and D^*_{02} designate the formal ***NCOA*** for

(MARRY, DIVORCE) and
(FUTURE, PAST),

respectively. (For $\boldsymbol{DIFF}(\text{TOM}, \text{BOB}) = 2$, see Ref. 3).) If no "*oppo*-knowledge" is available, $\boldsymbol{DIFF}^*(D_1, D_2)$ is only 38 (semit) (c.f. Fig. 1).

The idea we set a double score (of non-opposite case) to each opposite concept pair is that we think the semantic difference between two opposite concepts might be twice as large as the one between an irrelevant concept pair. This extension of the model makes it more adaptable to reality.

4. Generalization of $\boldsymbol{ELAB}$ to $\boldsymbol{SPEC}$

So far we have been discussing the natural language semantics by assuming elaboration relations. An elaboration relation gives rise to a strict reasoning from one concept to another. While, in natural language human often make reasoning by depending on ambiguous (non-strict) rules. So, we need to revise our basic model to a more flexible one to adapt to such reasoning.

⟨**Example 4-1**⟩

The following statements are admitted as true in human life, but they are not necessarily true in a strict sense:

(S1) Children like cookies.
(S2) People work for money.
(S3) If he is a Japanese, he eats raw fish.

These statements are equivalent to the following causality rules.

(R1) If X is a child, it is certain that X likes cookies.
(R2) If X is an ordinary person, X works for money.
(R3) If X is a Japanese, he may eat raw fish.

As we see in these examples, human can make probabilistic reasoning by using non-strict rules. In responding to those aspects of the natural language, we will extend the elaboration relation to a "specification relation (***SPEC***)" by introducing a new connector. "*indu*" is the new connector for it. The system knowledge takes the following form:

$(assu(D_2))indu(D_1)$
(If D_2, then probably D_1.)

For example, S3 above is described as follows.

$(assu([s(\text{X}), v(\text{BE}), c(\text{JAPANESE})]))$
$indu([s(\text{X}), v(\text{EAT}), o(\text{FISH/RAW})])$

If a concept (TARO) is unified with X in this rule, then we get an instantiated rule:

$(assu([s(\text{TARO}), v(\text{BE}), c(\text{JAPANESE})]))$
$indu([s(\text{TARO}), v(\text{EAT}), o(\text{FISH/RAW})])$
(If Taro is a Japanese, then he probably eats raw fish.)

In this case the following sentences become closer in meaning.

Taro is a Japanese.
Taro eats raw fish.

The specification relation between D_1 and D_2 is denoted by:

$\boldsymbol{SPEC}(D_1, D_2, n)$ or $\boldsymbol{SPEC}(D_1, D_2) = n$

Like ***ELAB*** relations, ***SPEC*** has two types, $\boldsymbol{SPEC}_{\boldsymbol{synt}}$ and $\boldsymbol{SPEC}_{\boldsymbol{know}}$. The formal definition of ***SPEC*** scores are as follows.

$$\begin{aligned}&\boldsymbol{SPEC}(D_1, D_2)\\ &\quad = min\{\boldsymbol{SPEC}_{\boldsymbol{synt}}(D_1, D_2),\\ &\qquad\quad \boldsymbol{SPEC}_{\boldsymbol{know}}(D_1, D_2)\},\end{aligned}$$

where

(A) $\boldsymbol{SPEC}_{\boldsymbol{synt}}(D_1, D_2) = \boldsymbol{ELAB}_{\boldsymbol{synt}}(D_1, D_2)$
(B) $\boldsymbol{SPEC}_{\boldsymbol{know}}(D_1, D_2) = \boldsymbol{ELAB}_{\boldsymbol{know}}(D_1, D_2)$
(C) $\boldsymbol{SPEC}_{\boldsymbol{know}}(D_1, D_2) = 3$,
if $(assu(D_2))indu(D_1)$ is a system knowledge.
(D) $\boldsymbol{SPEC}_{\boldsymbol{know}}(D_j, D_l/\text{MOST}) = 4$,
$\boldsymbol{SPEC}_{\boldsymbol{know}}(D_k, D_l/\text{MOST}) = 4$,
$\boldsymbol{SPEC}_{\boldsymbol{know}}(D_l/D, D_l/\text{MOST}) = 4$,
if $\boldsymbol{ELAB}_{\boldsymbol{know}}(D_l, D_j) = 2$
or $\boldsymbol{ELAB}_{\boldsymbol{know}}(D_l, D_k) = 3$ is secured.

These scores are illustrated in **Fig. 2**. They are all implemented in **SDENV-3**. As we see in this definition, ***ELAB*** is inherited to ***SPEC*** relation. An inference by a ***SPEC*** relation is

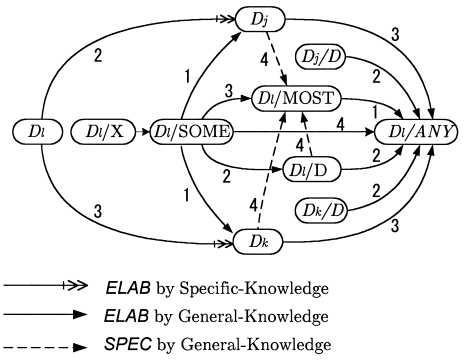

Fig. 2 General-knowledge based specification relations.

less reliable than an ***ELAB***.

Let us consider the following example:

⟨**Example 4-2**⟩

Let the system knowledge be

$$(\mathrm{ASIA})\mathit{incl}(\mathrm{JAPAN}),$$
$$(\mathit{assu}([s(\mathrm{TARO}), v(\mathrm{BE}), c(\mathrm{JAPANESE})]))$$
$$\mathit{indu}([s(\mathrm{TARO}), v(\mathrm{EAT}), o(\mathrm{FISH/RAW})])$$

In this case,

$$\boldsymbol{SPEC}(\mathrm{ASIA}, \mathrm{JAPAN}) = \boldsymbol{ELAB}(\mathrm{ASIA}, \mathrm{JAPAN}) = 3,$$
$$\boldsymbol{SPEC}([s(\mathrm{TARO}), v(\mathrm{BE}), c(\mathrm{JAPANESE})], [s(\mathrm{TARO}), v(\mathrm{EAT}), o(\mathrm{FISH/RAW})]) = 3,$$
$$\boldsymbol{SPEC}(\mathrm{JAPAN}, \mathrm{ASIA/MOST}) = 4,$$
$$\boldsymbol{SPEC}(\mathrm{ASIA/SOUTH}, \mathrm{ASIA/MOST}) = 4.$$

As far as the semantic difference score is concerned, we modify our definition of $\boldsymbol{DIFF}(D_1, D_2)$ by replacing all ***ELAB***'s with ***SPEC***'s.

5. Conclusions

The SD-Form Semantics Model is a framework for dealing with the semantics of natural language in a quantitative way. Details of the model specification are left open to the model users. The highlight of this model is that it allows us to compute a semantic difference measure between two concepts. The authors had already studied its possible applications through their experimental systems **SDENV-3**.

In the present article, we generalized $\boldsymbol{ELAB}(D_1, D_2, n)$ and $\boldsymbol{DIFF}(D_1, D_2)$ by introducing new ideas. We think that this generalization (extension) makes the model more adaptable to the real world.

References

1) Alshawi, H.: The Core Language Engine, *ACL-MIT Press Series in Natural Language Processing*, ACL-MIT Press (1992).
2) Shapiro, S.C.: *Encyclopedia of Computer Science*, Second Edition, John Wiley & Sons, New York (1992).
3) Wakiyama, M., Noda, H., Nozaki, K. and Kawaguchi, E.: Computation Algorithm of Semantic Difference Measure in the SD-Form Semantics Model, *Trans. IPS Japan*, Vol.40, No.3, pp.1065–1079 (1999).

(Received August 4, 1999)
(Accepted November 4, 1999)