GPSM: A GENERALIZED PROBABILISTIC SEMANTIC MODEL FOR AMBIGUITY RESOLUTION

†Jing-Shin Chang, ‡Yih-Fen Luo and †Keh-Yih Su

†Department of Electrical Engineering
National Tsing Hua University
Hsinchu, TAIWAN 30043, R.O.C.
†Email: shin@ee.nthu.edu.tw, kysu@ee.nthu.edu.tw

‡Behavior Design Corporation
No. 28, 2F, R&D Road II, Science-Based Industrial Park
Hsinchu, TAIWAN 30077, R.O.C.

ABSTRACT

In natural language processing, ambiguity resolution is a central issue, and can be regarded as a *preference* assignment problem. In this paper, a Generalized Probabilistic Semantic Model (GPSM) is proposed for preference computation. An effective *semantic tagging* procedure is proposed for tagging semantic features. A *semantic score* function is derived based on a *score function,* which integrates lexical, syntactic and semantic preference under a uniform formulation. The *semantic score* measure shows substantial improvement in structural disambiguation over a syntax-based approach.

1. Introduction

In a large natural language processing system, such as a machine translation system (MTS), ambiguity resolution is a critical problem. Various rule-based and probabilistic approaches had been proposed to resolve various kinds of ambiguity problems on a case-by-case basis.

In rule-based systems, a large number of rules are used to specify linguistic constraints for resolving ambiguity. Any parse that violates the semantic constraints is regarded as ungrammatical and rejected. Unfortunately, because every "rule" tends to have *exception* and *uncertainty,* and *ill-formedness* has significant contribution to the error rate of a large practical system, such "hard rejection" approaches fail to deal with these situations. A better way is to find all possible interpretations and place emphases on *preference*, rather than *well-formedness* (e.g., [Wilks 83].) However, most of the known approaches for giving preference depend heavily on heuristics such as counting the number of constraint satisfactions. Therefore, most such preference measures can not be objectively justified. Moreover, it is hard and costly to *acquire*, *verify* and *maintain* the consistency of the large fine-grained rule base by hand.

Probabilistic approaches greatly relieve the knowledge acquisition problem because they are usually *trainable*, *consistent* and easy to meet certain *optimum criteria*. They can also provide more objective preference measures for "soft rejection." Hence, they are attractive for a large system. The current probabilistic approaches have a wide coverage including lexical analysis [DeRose 88, Church 88], syntactic analysis [Garside 87, Fujisaki 89, Su 88, 89, 91b], restricted semantic analysis [Church 89, Liu 89, 90], and experimental translation systems [Brown 90]. However, there is still no *integrated* approach for modeling the joint effects of *lexical*, *syntactic* and *semantic* information on *preference* evaluation.

A generalized probabilistic semantic model (GPSM) will be proposed in this paper to overcome the above problems. In particular, an integrated formulation for lexical, syntactic and semantic knowledge will be used to derive the *semantic score* for semantic preference evaluation. Application of the model to structural disambiguation is investigated. Preliminary experiments show about 10%–14% improvement of the *semantic score* measure over a model that uses syntactic information only.

2. Preference Assignment Using Score Function

In general, a particular semantic interpretation of a sentence can be characterized by a set of *lexical categories* (or parts of speech), a *syntactic structure*, and the *semantic annotations* associated with it. Among the various interpretations of a sentence, the best choice should be the *most probable* semantic interpretation for the given input words. In other words, the interpretation that maximizes the following *score function* [Su 88, 89, 91b] or *analysis score* [Chen 91] is preferred:

$$
\begin{aligned}
& Score\,(Sem_i, Syn_j, Lex_k, Words) && (1)\\
& \equiv P\,(Sem_i, Syn_j, Lex_k | Words) && \\
& = P\,(Sem_i | Syn_j, Lex_k, Words) && \text{(semantic score)}\\
& \times\ P\,(Syn_j | Lex_k, Words) && \text{(syntactic score)}\\
& \times\ P\,(Lex_k | Words) && \text{(lexical score)}
\end{aligned}
$$

where (Lex_k, Syn_j, Sem_i) refers to the *k*th set of lexical categories, the *j*th syntactic structure and the *i*th set of semantic annotations for the input *Words*. The three component functions are referred to as *semantic score* (S_{sem}), *syntactic score* (S_{syn}) and *lexical score* (S_{lex}), respectively. The global preference measure will be referred to as *compositional score* or simply as *score*. In particular, the *semantic score* accounts for the semantic preference on a given set of lexical categories and a particular syntactic structure for the sentence. Various formulation for the lexical score and syntactic score had been studied extensively in our previous works [Su 88, 89, 91b, Chiang 92] and other literatures. Hence, we will concentrate on the formulation for *semantic score*.

3. Semantic Tagging

Canonical Form of Semantic Representation

Given the formulation in Eqn. (1), first we will show how to extract the abstract objects (Sem_i, Syn_j, Lex_k) from a semantic representation. In general, a particular interpretation of a sentence can be represented by an *annotated syntax tree* (AST), which is a syntax tree annotated with *feature structures* in the tree nodes. Figure 1 shows an example of AST. The *annotated* version of a node A is denoted as $\overline{A} \equiv A\,[f_A]$ in the figure, where f_A is the feature structure associated with node A. Because an AST preserves both syntactic and semantic information, it can be converted to other deep structure representations easily. Therefore, without lose of generality, the AST representation will be used as the canonical form of semantic representation for preference evaluation. The techniques used here, of course, can be applied to other deep structure representations as well.

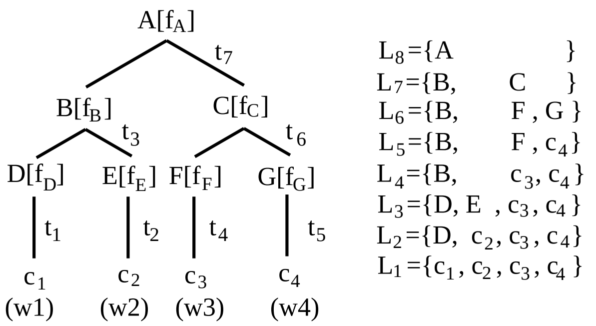

Figure 1. Annotated Syntax Tree (AST) and Phrase Levels (PL).

The hierarchical AST can be represented by a set of *phrase levels*, such as L_1 through L_8 in Figure 1. Formally, a *phrase level* (PL) is a set of symbols corresponding to a *sentential form* of the sentence. The phrase levels in Figure 1 are derived from a sequence of *rightmost derivations*, which is commonly used in an LR parsing mechanism. For example, L_5 and L_4 correspond to the rightmost derivation $B\ F\ c_4 \overset{+}{\underset{rm}{\Rightarrow}} B\ c_3\ c_4$. Note that the first phrase level L_1 consists of all lexical categories c_1 ... c_n of the terminal words (w_1 ... w_n). A phrase level with each symbol annotated with its feature structure is called an *annotated phrase level* (APL). The *i-th* APL is denoted as Γ_i. For example, L_5 in Figure 1 has an annotated phrase level $\Gamma_5 = \{B\,[f_B]\,, F\,[f_F]\,, c_4\,[f_{c_4}]\}$ as its counterpart, where f_{c_4} is the atomic feature of the lexical category c_4, which comes from the lexical item of the *4th* word w_4. With the above notations, the score function can be re-formulated as follows:

$$
\begin{aligned}
&Score\,(Sem_i, Syn_j, Lex_k, Words) && (2)\\
&\quad \equiv P\left(\Gamma_1^m, L_1^m, c_1^n | w_1^n\right) &&\\
&\quad = P\left(\Gamma_1^m | L_1^m, c_1^n, w_1^n\right) && (\text{semantic score})\\
&\quad \times P\left(L_1^m | c_1^n, w_1^n\right) && (\text{syntactic score})\\
&\quad \times P\left(c_1^n | w_1^n\right) && (\text{lexical score})
\end{aligned}
$$

where c_1^n (a short form for $\{c_1 \ldots c_n\}$) is the *k*th set of lexical categories (Lex_k), L_1^m ($\{L_1 \ldots L_m\}$) is the *j*th syntactic structure (Syn_j), and Γ_1^m ($\{\Gamma_1 \ldots \Gamma_m\}$) is the *i*th set of semantic annotations (Sem_i) for the input words w_1^n ($\{w_1 \ldots w_n\}$). A good encoding scheme for the Γ_i's will allow us to take semantic information into account without using redundant information. Hence, we will show how to annotate a syntax tree so that various interpretations can be characterized differently.

Semantic Tagging

A popular linguistic approach to annotate a tree is to use a unification-based mechanism. However, many information irrelevant to disambiguation might be included. An effective encoding scheme should be simple yet can preserve most *discrimination information* for disambiguation. Such an encoding scheme can be accomplished by associating each phrase structure rule $A \rightarrow X_1 X_2 \ldots X_M$ with a *head list* $(X_{i_1}, X_{i_2} \ldots X_{i_M})$. The head list is formed by arranging the children nodes $(X_1, X_2, \ldots, X_M)$ in *descending* order of importance to the compositional semantics of their mother node A. For this reason, X_{i_1}, X_{i_2} and X_{i_j} are called the *primary*, *secondary* and the *j-th heads* of A, respectively. The compositional semantic features of the mother node A can be represented as an ordered list of the feature structures of its children, where the order is the same as in the head list. For example, for S $\rightarrow$ NP VP, we have a head list (VP, NP), because VP is the (primary) head of the sentence. When composing the compositional semantics of S, the features of VP and NP will be placed in the first and second slots of the feature structure of S, respectively.

Because not all children and all features in a feature structure are equally significant for disambiguation, it is not really necessary to annotate a node with the feature *structures* of *all* its children. Instead, only the most important N children of a node is needed in characterizing the node, and only the most *discriminative* feature of a child is needed to be passed to its mother node. In other words, an N-dimensional feature vector, called a *semantic N-tuple*, could be used to characterize a node without losing much information for disambiguation. The *first* feature in the semantic N-tuple comes from the *primary* head, and is thus called the *head feature* of the semantic N-tuple. The other features come from the other children in the order of the head list. (Compare these notions with the linguistic sense of *head* and *head feature*.) An annotated node can thus be approximated as $\overline{A} \approx A\,(f_1, f_2, \cdots, f_N)$, where $f_j = HeadFeature\left(\overline{X_{i_j}}\right)$ is the (primary) *head feature* of its *j-th* head (i.e., X_{i_j}) in the head list. Non-head features of a child node X_{i_j} will not be percolated up to its mother node. The head feature of $\overline{A}$ itself, in this case, is f_1. For a *terminal* node, the head feature will be the *semantic tag* of the corresponding lexical item; other features in the N-tuple will be tagged as ϕ (NULL).

Figure 2 shows two possible annotated syntax trees for the sentence "... saw the boy in the park." For instance, the "loc(ation)" feature of "park" is percolated to its mother NP node as the head feature; it then serves as the secondary head feature of its grandmother node PP, because the NP node is the secondary head of PP. Similarly, the VP node in the left tree is annotated as VP(sta,anim) according to its primary head saw(sta,ϕ) and secondary head NP(anim,in). The VP(sta,in) node in the right tree is tagged differently, which reflects different attachment preference of the prepositional phrase.

By this simple mechanism, the major characteristics of the children, namely the head features, can be percolated to higher syntactic levels, and

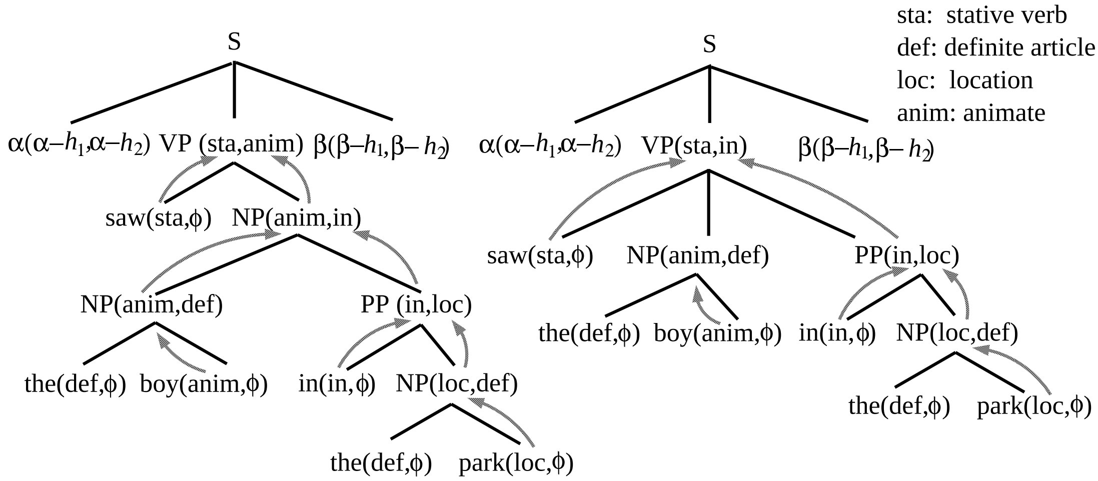

Figure 2. Ambiguous PP attachment patterns annotated with semantic 2–tuples.

their correlation and dependency can be taken into account in preference evaluation even if they are far apart. In this way, different interpretations will be tagged differently. The preference on a particular interpretation can thus be evaluated from the distribution of the annotated syntax trees. Based on the above semantic tagging scheme, a *semantic score* will be proposed to evaluate the semantic preference on various interpretations for a sentence. Its performance improvement over *syntactic score* [Su 88, 89, 91b] will be investigated. Consequently, a brief review of the syntactic score evaluation method is given before going into details of the semantic score model. (See the cited references for details.)

4. Syntactic Score

According to Eqn. (2), the *syntactic score* can be formulated as follows [Su 88, 89, 91b]:

$$
\begin{aligned}
S_{syn} &\equiv P\left(Syn_j | Lex_k, w_1^n\right) = P\left(L_1^m | c_1^n, w_1^n\right) \qquad (3)\\
&= \prod_{l=2}^{m} P\left(L_l | L_1^{l-1}, c_1^n, w_1^n\right)\\
&\approx \prod P\left(L_l | L_1^{l-1}\right)\\
&\approx \prod P\left(L_l | L_{l-1}\right)\\
&= \prod P\left(\{\alpha_l, A_l, \beta_l\} \,|\, \{\alpha_l, X_1, X_2, \cdots, X_M, \beta_l\}\right)
\end{aligned}
$$

where α_l, β_l are the left context and right context under which the derivation $A_l \stackrel{+}{\Rightarrow} X_1 X_2 \ldots X_M$ occurs. (Assume that $L_l = \{\alpha_l, A_l, \beta_l\}$ and $L_{l-1} = \{\alpha_l, X_1, \cdots, X_M, \beta_l\}$.) If L left context symbols in α_l and R right context symbols in β_l are consulted to evaluate the syntactic score, it is said to operate in $L_L R_R$ *mode* of operation. When the context is ignored, such an $L_0 R_0$ mode of operation reduces to a stochastic *context-free* grammar.

To avoid the *normalization problem* [Su 91b] arisen from different number of transition probabilities for different syntax trees, an alternative formulation of the syntactic score is to evaluate the transition probabilities between configuration changes of the parser. For instance, the configuration of an LR parser is defined by its stack contents and input buffer. For the AST in Figure 1, the parser configurations after the read of c_1, c_2, c_3, c_4 and \$ (end-of-sentence) are equivalent to L_1, L_2, L_4, L_5 and L_8, respectively. Therefore, the syntactic score can be approximated as [Su 89, 91b]:

$$
\begin{aligned}
S_{syn} &\approx P\left(L_8, L_7 \cdots L_2 | L_1\right) \qquad (4)\\
&\approx P\left(L_8 | L_5\right) \times P\left(L_5 | L_4\right) \times P\left(L_4 | L_2\right) \times P\left(L_2 | L_1\right)
\end{aligned}
$$

In this way, the number of transition probabilities in the syntactic scores of all AST's will be kept the same as the sentence length.

5. Semantic Score

Semantic score evaluation is similar to syntactic score evaluation. From Eqn. (2), we have the following semantic model for *semantic score*:

$$
\begin{aligned}
& S_{sem}\left(Sem_i, Syn_j, Lex_k, Words\right) \qquad (5) \\
& \equiv P\left(\Gamma_1^m \mid L_1^m, c_1^n, w_1^n\right) \\
& = \prod_{l=2}^{m} P\left(\Gamma_l \mid \Gamma_1^{l-1}, L_1^m, c_1^n, w_1^n\right) \\
& \approx \prod P\left(\Gamma_l \mid \Gamma_{l-1}\right) \\
& = \prod P\left(\{\overline{\alpha}_l, \overline{A}_l, \overline{\beta}_l\} \mid \{\overline{\alpha}_l, \overline{X_1}, \overline{X_2}, \cdots, \overline{X_M}, \overline{\beta}_l\}\right)
\end{aligned}
$$

where $\overline{A_l} \equiv A_l\left(f_{l,1}, f_{l,2}, \cdots, f_{l,N}\right)$ is the annotated version of A_l, whose semantic N-tuple is $(f_{l,1}, f_{l,2}, \cdots, f_{l,N})$, and $\overline{\alpha_l}$, $\overline{\beta_l}$ are the annotated context symbols. Only Γ_{l-1} is assumed to be significant for the transition to Γ_l in the last equation, because all required information is assumed to have been percolated to Γ_{l-1} through semantics composition.

Each term in Eqn. (5) can be interpreted as the probability that A_l is annotated with the particular set of head features $(f_{l,1}, f_{l,2}, \cdots, f_{l,N})$, given that X_1 ... X_M are reduced to A_l in the context of $\overline{\alpha_l}$ and $\overline{\beta_l}$. So it can be interpreted informally as $P(A_l\left(f_{l,1}, f_{l,2}, \cdots, f_{l,N}\right) \mid A_l \leftarrow X_1 \cdots X_M,$ $\text{in the context of } \overline{\alpha_l}, \overline{\beta_l})$. It corresponds to the semantic preference assigned to the annotated node $\overline{A}_l$. Since $(f_{l,1}, f_{l,2}, \cdots f_{l,N})$ are the head features from various heads of the substructures of A, each term reflects the *feature co-occurrence* preference among these heads. Furthermore, the heads could be very far apart. This is different from most simple Markov models, which can deal with local constraints only. Hence, such a formulation well characterizes long distance dependency among the heads, and provides a simple mechanism to incorporate the feature co-occurrence preference among them. For the semantic N-tuple model, the semantic score can thus be expressed as follows:

$$
\begin{aligned}
& S_{sem} \qquad (6) \\
& \approx \prod_{l=2}^{m} P\left(A_l\left(f_{l,1}, f_{l,2} \cdots f_{l,N}\right) \mid \alpha_l, A_l \leftarrow X_1 \cdots X_M, \beta_l\right)
\end{aligned}
$$

where $f_{l,j}$ are the semantic tags from the children of A_l. For example, we have terms like $P\left(\mathrm{VP}\left(\mathrm{sta}, \mathrm{anim}\right) \mid \alpha, \mathrm{VP} \leftarrow \mathrm{v\ \ NP}, \beta\right)$ and $P\left(\mathrm{VP}\left(\mathrm{sta}, \mathrm{in}\right) \mid \alpha, \mathrm{VP} \leftarrow \mathrm{v\ \ NP\ \ PP}, \beta\right)$, respectively, for the left and right trees in Figure 2. The annotations of the context are ignored in evaluating Eqn. (6) due to the assumption of semantics compositionality. The operation mode will be called $L_L R_R$+A_N, where N is the dimension of the N-tuple, and the subscript L (or R) refers to the size of the context window. With an appropriate N, the score will provide sufficient discrimination power for general disambiguation problem without resorting to full-blown semantic analysis.

6. Major Categories and Semantic Features

As mentioned before, not all constituents are equally important for disambiguation. For instance, *head words* are usually more important than *modifiers* in determining the compositional semantic features of their mother node. There is also lots of redundancy in a sentence. For instance, “saw boy in park” is equally recognizable as “saw the boy in the park.” Therefore, only a few categories, including *verbs*, *nouns*, *adjectives*, *prepositions* and *adverbs* and their projections (NP, VP, AP, PP, ADVP), are used to carry semantic features for disambiguation. These categories are roughly equivalent to the *major categories* in linguistic theory [Sells 85] with the inclusion of *adverbs* as the only difference.

The semantic feature of each major category is encoded with a set of *semantic tags* that well describes each category. A few rules of thumb are used to select the semantic tags. In particular, semantic features that can discriminate different linguistic behavior from different possible semantic *N-tuples* are preferred as the semantic tags. With these heuristics in mind, the verbs, nouns, adjectives, adverbs and prepositions are divided into 22, 30, 14, 10 and 28 classes, respectively. For example, the nouns are divided into “human,” “plant,” “time,” “space,” and so on. These semantic classes come from a number of sources and the semantic attribute hierarchy of the **ArchTran** MTS [Su 90, Chen 91].

7. Test and Analysis

The *semantic N-tuple* model is used to test the improvement of the *semantic score* over *syntactic score* in structure disambiguation. Eqn. (3) is adopted to evaluate the *syntactic* score in L_2R_1 mode of operation. The *semantic* score is derived from Eqn. (6) in $L_2R_1+A_N$ mode, for N = 1, 2, 3, 4, where N is the dimension of the semantic N-tuple.

A total of 1000 sentences (including 3 unambiguous ones) are randomly selected from 14 computer manuals for training or testing. They are divided into 10 parts; each part contains 100 sentences. In *close tests*, 9 parts are used both as the training set and the testing set. In *open tests*, the *rotation estimation* approach [Devijver 82] is adopted to estimate the open test performance. This means to iteratively test *one* part of the sentences while using the remaining parts as the training set. The overall performance is then estimated as the average performance of the 10 iterations.

The performance is evaluated in terms of *Top-N recognition rate* (TNRR), which is defined as the fraction of the test sentences whose preferred interpretation is successfully ranked in the first N candidates. Table 1 shows the simulation results of *close tests*. Table 2 shows partial results for *open tests* (up to rank 5.) The recognition rates achieved by considering *syntactic score* only and *semantic score* only are shown in the tables. ($L_2R_1+A_3$ and $L_2R_1+A_4$ performance are the same as $L_2R_1+A_2$ in the present test environment. So they are not shown in the tables.) Since each sentence has about 70–75 ambiguous constructs on the average, the task perplexity of the current disambiguation task is high.

Table 1. Close Test of Semantic Score

Score	Syntax (L2R1)		Semantics (L2R1+A1)		Semantics (L2R1+A2)	
Rank	Count	TNRR (%)	Count	TNRR (%)	Count	TNRR (%)
1	781	87.07	872	97.21	866	96.54
2	101	98.33	20	99.44	24	99.22
3	9	99.33	5	100.00	4	99.67
4	5	99.89			-	-
5	-	-			2	99.89
13	-	-			1	100.00
18	1	100.00				
DataBase: 900 Sentences						
Test Set: 897 Sentences						
Total Number of Ambiguous Trees = 63233						

(*) TNRR: Top-N Recognition Rate

Table 2. Open Test of Semantic Score

Score	Syntax (L2R1)		Semantics (L2R1+A1)		Semantics (L2R1+A2)	
Rank	Count	TNRR (%)	Count	TNRR (%)	Count	TNRR (%)
1	430	43.13	569	57.07	578	57.97
2	232	66.40	163	73.42	167	74.72
3	94	75.83	90	82.45	75	82.25
4	80	83.85	50	87.46	49	87.16
5	35	87.36	22	89.67	28	89.97
DataBase: 900 Sentences (+)						
Test Set: 997 Sentences (++)						
Total Number of Ambiguous Trees = 75339						

(+) DataBase : effective database size for rotation estimation

(++) Test Set : all test sentences participating the rotation estimation test

The close test Top-1 performance (Table 1) for *syntactic* score (87%) is quite satisfactory. When *semantic* score is taken into account, substantial improvement in recognition rate can be observed further (97%). This shows that the semantic model does provide an effective mechanism for disambiguation. The recognition rates in open tests, however, are less satisfactory under the present test environment. The open test performance can be attributed to the small database size and the estimation error of the parameters thus introduced. Because the training database is small with respect to the complexity of the model, a significant fraction of the probability entries in the testing set can not be found in the training set. As a result, the parameters are somewhat "overtuned" to the training database, and their values are less favorable for open tests. Nevertheless, in both close tests and open tests, the semantic score model shows substantial improvement over syntactic score (and hence stochastic context-free grammar). The improvement is about 10% for close tests and 14% for open tests.

In general, by using a larger database and better robust estimation techniques [Su 91a, Chiang 92], the baseline model can be improved further. As we had observed from other experiments for spoken language processing [Su 91a], lexical tagging, and structure disambiguation [Chiang 92], the performance under sparse data condition can be improved significantly if robust adaptive learning techniques are used to adjust the initial parameters. Interested readers are referred to [Su 91a, Chiang 92] for more details.

8. Concluding Remarks

In this paper, a *generalized probabilistic semantic model* (GPSM) is proposed to assign *semantic preference* to ambiguous interpretations. The semantic model for measuring preference is based on a score function, which takes lexical, syntactic and semantic information into consideration and optimizes the joint preference. A simple yet effective encoding scheme and semantic tagging procedure is proposed to characterize various interpretations in an N dimensional feature space. With this encoding scheme, one can encode the interpretations with discriminative features, and take the feature co-occurrence preference among various constituents into account. Unlike simple Markov models, long distance dependency can be managed easily in the proposed model. Preliminary tests show substantial improvement of the semantic score measure over syntactic score measure. Hence, it shows the possibility to overcome the ambiguity resolution problem without resorting to full-blown semantic analysis.

With such a simple, objective and trainable formulation, it is possible to take high level semantic knowledge into consideration in statistic sense. It also provides a *systematic* way to construct a disambiguation module for large practical machine translation systems without much human intervention; the heavy burden for the linguists to write fine-grained "rules" can thus be relieved.

REFERENCES

[Brown 90] Brown, P. et al., "A Statistical Approach to Machine Translation," *Computational Linguistics*, vol. 16, no. 2, pp. 79-85, June 1990.

[Chen 91] Chen, S.-C., J.-S. Chang, J.-N. Wang and K.-Y. Su, "ArchTran: A Corpus-Based Statistics-Oriented English-Chinese Machine Translation System," *Proceedings of Machine Translation Summit III*, pp. 33–40, Washington, D.C., USA, July 1–4, 1991.

[Chiang 92] Chiang, T.-H., Y.-C. Lin and K.-Y. Su, "Syntactic Ambiguity Resolution Using A Discrimination and Robustness Oriented Adaptive Learning Algorithm", to appear in *Proceedings of COLING-92*, 14th Int. Conference on Computational Linguistics, Nantes, France, 20–28 July, 1992.

[Church 88] Church, K., "A Stochastic Parts Program and Noun Phrase Parser for Unrestricted Text," *ACL Proc. 2nd Conf. on Applied Natural Language Processing*, pp. 136-143, Austin, Texas, USA, 9-12 Feb. 1988.

[Church 89] Church, K. and P. Hanks, "Word Association Norms, Mutual Information, and Lexicography," *Proc. 27th Annual Meeting of the ACL*, pp. 76-83, University of British Columbia, Vancouver, British Columbia, Canada, 26-29 June 1989.

[DeRose 88] DeRose, Steven. J., "Grammatical Category Disambiguation by Statistical Optimization," *Computational Linguistics*, vol. 14, no. 1, pp. 31-39, 1988.

[Devijver 82] Devijver, P.A., and J. Kittler, *Pattern Recognition: A Statistical Approach*, Prentice-Hall, London, 1982.

[Fujisaki 89] Fujisaki, T., F. Jelinek, J. Cocke, E. Black and T. Nishino, "A Probabilistic Parsing Method for Sentence Disambiguation," *Proc. of Int. Workshop on Parsing Technologies* (IWPT-89), pp. 85–94, CMU, Pittsburgh, PA, U.S.A., 28-31 August 1989.

[Garside 87] Garside, Roger, Geoffrey Leech and Geoffrey Sampson (eds.), *The Computational Analysis of English: A Corpus-Based Approach*, Longman Inc., New York, 1987.

[Liu 89] Liu, C.-L., *On the Resolution of English PP Attachment Problem with a Probabilistic Semantic Model*, Master Thesis, National Tsing Hua University, Hsinchu, TAIWAN, R.O.C., 1989.

[Liu 90] Liu, C.-L, J.-S. Chang and K.-Y. Su, "The Semantic Score Approach to the Disambiguation of PP Attachment Problem," *Proc. of ROCLING-III*, pp. 253-270, Taipei, R.O.C., September 1990.

[Sells 85] Sells, Peter, *Lectures On Contemporary Syntactic Theories: An Introduction to Government-Binding Theory, Generalized Phrase Structure Grammar, and Lexical-Functional Grammar,* CSLI Lecture Notes Number 3, Center for the Study of Language and Information, Leland Stanford Junior University., 1985.

[Su 88] Su, K.-Y. and J.-S. Chang, "Semantic and Syntactic Aspects of Score Function," *Proc. of COLING-88*, vol. 2, pp. 642-644, 12th Int. Conf. on Computational Linguistics, Budapest, Hungary, 22-27 August 1988.

[Su 89] Su, K.-Y., J.-N. Wang, M.-H. Su and J.-S. Chang, "A Sequential Truncation Parsing Algorithm Based on the Score Function," *Proc. of Int. Workshop on Parsing Technologies* (IWPT-89), pp. 95-104, CMU, Pittsburgh, PA, U.S.A., 28-31 August 1989.

[Su 90] Su, K.-Y. and J.-S. Chang, "Some Key Issues in Designing MT Systems," *Machine Translation*, vol. 5, no. 4, pp. 265-300, 1990.

[Su 91a] Su, K.-Y., and C.-H. Lee, "Robustness and Discrimination Oriented Speech Recognition Using Weighted HMM and Subspace Projection Approach," *Proceedings of IEEE ICASSP-91*, vol. 1, pp. 541-544, Toronto, Ontario, Canada. May 14-17, 1991.

[Su 91b] Su, K.-Y., J.-N. Wang, M.-H. Su, and J.-S. Chang, “GLR Parsing with Scoring”. In M. Tomita (ed.), *Generalized LR Parsing*, Chapter 7, pp. 93–112, Kluwer Academic Publishers, 1991.

[Wilks 83] Wilks, Y. A., "Preference Semantics, Ill-Formedness, and Metaphor," *AJCL*, vol. 9, no. 3-4, pp. 178 - 187, July - Dec. 1983.