Using frames in Spoken Language Understanding

Renato De Mori

Laboratoire d'Informatique – Université d'Avignon - France
renato.demori@univ-avignon.fr

Abstract

This paper reviews basic concepts for natural spoken language interpretation by computers. Frame structures are described as suitable computer representations of semantic compositions. A process is introduced for obtaining basic semantic constituents by translating word sequences into basic semantic constituents and for composing constituent hypotheses into frame structures. Experimental results with the French telephone corpora are reported. They show that Finite State conceptual language models are useful for translating word hypotheses into states representing progressive semantic compositions and the use of Conditional Random Fields (CRF) improves the accuracy of constituent hypothesization.

Index Terms: Spoken Language Understanding, computer meaning representation, meaning representation languages, Frames, finite-state conceptual language models.

1. Introduction

Epistemology, the science of knowledge, considers a datum as basic unit. A datum can be an object, an action or an event in the world and can have time and space coordinates, multiple aspects and qualities that make it different from others. A datum can be represented by a word or it can be abstract and be represented by a concept. There may be relations among data.

Computer epistemology deals with observable facts and their representation in a computer. Knowledge about the structure of a domain represents a datum by an object and groups objects into *classes* by their properties. Classes are organized into *hierarchies.* An object is an *instance* of a class. Judgment is expressed by *predicates* which describe. Predicates have arguments which are variables whose values have to respect some constraints.

Natural language refers to data in the world and their relations. Sentences of a natural language are sequences of words. Groups of words have associated conceptualizations also called *meanings* which can be selected and composed to form the meaning of the sentence.

Semantics deals with the organization of meanings and the relations between signs or symbols and what they denote or mean (Woods, 1975). Human conceptualization of the world is not well understood. Nevertheless, good models for this organization assume that basic *semantic constituents* expressed by a language are organized into *conceptual structures*.

In (Jackendoff, 2002, p. 124) it is suggested that semantics is an independent generative system correlated with syntax through an interface. Computer semantics performs a conceptualization of the world using well defined elements of programming languages. Programming languages have their own syntax and semantic. The former defines legal programming statements, the latter specifies the operations a machine performs when an instruction is executed. Specifications are defined in terms of the procedures the machine has to carry out. Semantic analysis of a computer program is essential for understanding the behavior of a program and its coherence with the design concepts and goals.

Natural language interpretation by computers generate concept hypotheses represented in a *semantic language*. The definition of a semantic language can be based on a formal grammar but has to include procedures for obtaining interpretations from sentences. Procedures are executed by computational processes belonging to an *interpretation strategy*.

Computer programs conceived for interpreting natural language differ from the human process they model. They can be considered as approximate models for developing useful applications, interesting research experiments and demonstrations. Semantic representations in computers usually treat data as *objects* respecting logical *adequacy* in order to formally represent any particular interpretation of a sentence. Even if utterances, in general, convey meanings which may not have relations which can be expressed in formal logic (Jackendoff, 2002, p. 287), formal logic has been considered adequate for representing natural language semantics in many application domains.

In many applications, computer systems interpret natural language for performing actions such as a data base access and display of the results and may require the use of knowledge which is not coded into the sentence but can be inferred from the system knowledge stored in long or short term memories.

It is argued in (Woods, 1975) that a specification for natural language semantics requires more than the transformation of a sentence into a representation. In fact, computer representations should permit, among other things, legitimate conclusions to be drawn from data (Mc Carty and Hayes, 1969).

Spoken Language Understanding (SLU) is the interpretation of signs conveyed by a speech signal. This is a difficult task because meaning is mixed with other information like speaker identity and environment. Natural language sentences are often difficult to parse and spoken messages are often ungrammatical. The knowledge used is often imperfect and the transcription of user utterances in terms of word hypotheses is performed by an Automatic Speech Recognition (ASR) system which makes errors.

Some important challenges in SLU are:

- meaning representation,

- definition and representation of signs,
- conception of relations between signs and meaning and between instances of meaning,
- processes for sign extraction, generation of hypotheses about units of meaning and constituent composition into semantic structures,
- robustness and evaluation of confidence for semantic hypotheses,
- automatic learning of relations from annotated corpora,
- collection and semantic annotation of corpora.

This paper describes a process for SLU. Reviews on SLU research can be found in in (De Mori, 1998, Wang 2006 and Mc Tear 2006).

2. Computer representations of meaning using frames

Computer representation of meaning is described by a Meaning Representation Language (MRL). It is preferable that MRL is conceived with reference to a representation model coherent with a theory of epistemology. As such, it should take into account, *intension* and *extension*, relations, reasoning, composition of semantic constituents into structures, procedures for relating them with signs.

The semantic knowledge of an application is a *knowledge base (KB).* A convenient way for reasoning about semantic knowledge is to represent it as a set of logic formulas. Formulas contain variables which are bound by constants and may be typed. An object is built by binding all the variables of a formula or by composing existing objects.

Semantic compositions and decisions about composition actions are the result of an inference process. Basic *inference problem* is to determine whether $\mathrm{KB} \models \mathrm{F}$ which means that KB *entails* a formula F, meaning that F is true in all possible variable assignments (worlds) for which KB is true.

The formulas in a KB describe concepts and their relations which can be represented in a network called *semantic network*. A semantic network is made of nodes corresponding to entities and links corresponding to relations. This model combines the ability to store factual knowledge and to model associative connections between entities (Woods, 1975).

The structure of semantic networks can be defined by a graph grammar. Computer programming classes and objects called *frames* can be defined to represent entities and relations in semantic networks. Frame representation can be derived from semantic networks They are computational structures (Kifer et al., 1995) and also cognitive structuring devices in a semantic construction theory (Fillmore, 1968).

Part of a frame is a data structure which represents a concept by associating to the concept name a set of roles which are represented by *slots*. Finding values for roles corresponds to fill the frame slots. A *slot filler* can be the instance of another frame. There may be *necessary* and *optional* slots. *Fillers* can be obtained by *attachment* of procedures or detectors (of e.g. noun groups), *inheritance*, default.

A *facets* can be associated to a slot. Constraints on the values that can fill a slot can be stored into a slot facet. Constraints can be expressed by probability distributions on the possible filler values (Koller, 1998).

Descriptions are attached to slots to specify constraints. Descriptions may have connectives, coreferential (descriptions attached to a slot are attached to another and vice-versa), declarative conditions.

Verbs are fundamental components of natural language sentences. They represent actions for which different entities play different roles. Actions reveal how sentence phrases and clauses are semantically related to verbs by expressing cases for verbs. A *case* is the name of a particular *role* that a noun phrase or other component takes in the state or activity expressed by the *verb* in a sentence. There is a case structure for each main verb. Attempts were made for mapping specific *surface cases* into a deep semantic representation expressing a sort of semantic invariant. Many deep semantic representations are based on *deep case n-ary relations* between concepts as proposed by Fillmore (Fillmore, 1968). *Deep case* systems have very few cases each one representing a basic semantic constraint.

Early frame representations were used to represent facts about an object with a property list. For example, a specific address can be represented by the following frame:

{a0001

instance_of	address
loc	Avignon
area	Vaucluse
country	France
street	1, avenue Pascal
zip	84000}

Here a0001 is a handle that represents an instance of a class which is specified by the value of the first slot. The other slots, made of a property name and a value, define the property list of this particular instance of the class “address”.

The above frame can be derived (Nilsson, 1981), after skolemization from the following logic formula:

$$(\exists x)\left\{\begin{array}{l}\mathrm{instance\_of}(x,\mathrm{address}) \wedge \mathrm{loc}(x,\mathrm{Avignon}) \wedge \\ \wedge\ \mathrm{area}(x,\mathrm{Vaucluse}) \wedge \mathrm{country}(x,\mathrm{France}) \wedge \\ \wedge\ \mathrm{street}(x,1\ \mathrm{avenue\ Pascal}) \wedge \mathrm{zip}(x,84000)\end{array}\right\}$$

A definition, with a similar syntax, but with a different semantic is provided for the address class which defines the structure of any address:

{address

loc	TOWN
area	DEPARTMENT OR PROVINCE OR STATE
country	NATION
street	NUMBER AND NAME
zip	ORDINAL NUMBER}

The syntactic analysis of a parsable sentence can be used for establishing relations between syntactic structures and meaning. Concerning the relation between syntax and semantics, in (Jackendoff, 1990), it is observed that:

- Each major syntactic constituent of a sentence maps into a conceptual constituent, but the inverse is not true.
- Each conceptual constituent supports the encoding of units (linguistic, visual,…).
- Many of the categories support *type|token* distinction.
- Many of the categories support quantification.
- Some realizations of conceptual categories in conceptual structures can be decomposed into a *function|argument* structure.

For certain types of applications, domain-dependent semantic knowledge has been integrated into stochastic semantic grammar. A survey on these grammars and their use can be found in (Wang, 2005)

3. Conceptual language models for a modular SLU architecture

Generation of hypotheses about semantic constituents and semantic composition are different operations in nature and can be performed by different techniques implemented in different modules. Each module can integrate different models in order to improve robustness. Specific *conceptual language models* can be used in ASR decoding to obtain constituent hypotheses directly from the signal or from word hypotheses. Other types of knowledge are used in shallow parsers(Pradhan, 2004). In order to avoid the complexity of context-free and context-sensitive grammars, finite-state approximations of context-free grammars are proposed in (Pereira, 1990). Approximations of TAG grammars are described in (Rambow et al., 2002). A review of these approximations is provided in (Erdogan et al., 2005).

In both cases, a generic n-gram LM can be used with specific stochastic finite-state machines (FSM), one for each semantic constituent c_j.An example of LMs based on stochastic FSMs can be found in (Prieto et al., 1994). Stochastic Automata and their use for hypothesizing semantic constituents are proposed in (Gorin 1997, Nasr., 1999). Finite-state Hidden Markov Models (HMM) for SLU are proposed in (Pieraccini, 1991).

In (Kawahara et al., 1999), an automaton extracts key phrases from continuous speech and converts them to commands for a multi-modal interaction with a virtual fitting room. Interpolation of generic n-gram models and specific concept models is performed by maximizing the divergence between a linear interpolation of the two models and the generic n-gram model. A greedy algorithm is proposed (Riccardi and Gorin, 2000).

In (Drenth and Ruber, 1997), it is proposed to obtain a semantic interpretation of a dialog "turn" (one or more sentences) by extracting concept hypotheses from a word lattice. Each concept hypothesis is extracted with a *conceptual semantic context-free grammar*.

Finite state models can be made more robust by modifying the original topology to take into account possible insertions, deletions and substitutions. Insertion of words not essential for characterizing a semantic constituent can be modeled by groups of syllables.

Recent advances in research on stochastic FSM made it possible to generate a probabilistic lattice of conceptual constituent hypotheses from a probabilistic lattice of word hypotheses.

The solution proposed in (Raymond et al., 2006) is now introduced. A stochastic finite-state conceptual language model CLM_j is conceived for every semantic constituent c_j. An initial ASR activity uses a generic LM, indicated as GENLM, for generating a graph of word hypotheses. Let WG be the stochastic FSM representing the lattice of word hypotheses generated by an ASR system. A knowledge source, is built by connecting all the CLM_j in parallel as shown in Figure 1. Such a knowledge source is composed with WG leading to an automaton SEMG in which concept tags representing semantic constituents are added to arcs in WG:

$$SEMG = WG \circ \left(\bigcup_{c=0}^{C} CLM_c \right)$$

operator $\circ$ indicates composition.

CLM_0 is a generic model for sequences of words which do not express concepts in the application domain.

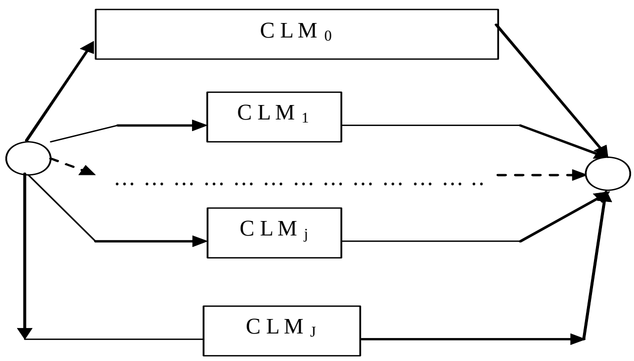

Figure 1 – Composition of conceptual language models

In order to obtain the concept tags representing hypotheses that are more likely to be expressed by the analyzed utterance, SEMG is projected on its outputs leading to a weighted Finite State Machine (FSM) with only indicators of beginning and end words of semantic tags. The resulting FSM is then made deterministic and minimized leading to an FSM SWG given by:

SWG=OUTPROJ(SEMG)

where OUTPROJ represents the operation of projection on the outputs followed by determinization and minimization.

A sequential interpretation strategy for a dialogue service in the France Telecom 3000 system (Minescu, 2007) using confusion networks (Hakkani-Tur, 2006) on relevant messages only. Results are reported in Table 1.

Conditional Random Fields (CRF) have been used for generating hypotheses about semantic constituents for the

MEDIA French corpus. Results and comparisons with other methods obtained by (Raymond, personal communication, 2007) on predicate/attribute pairs using the 1 best ASR hypothesis are provided in Table 2. Further 10% error reduction have been observed by method combination.

Table 1 – Interpretation results using conceptual LMs

	Baseline (1-best)	*sequential strategy*
Insertion rate	17.2 %	8.8 %
Substitution rate	6.1 %	5.6 %
Deletion rate	2.7 %	5.2 %
Interpretation error rate (IER)	**26.0 %**	**19.6 %**

Table 2 – Comparison of interpretation results obtained in the MEDIA corpus

	concept error rate (CER)
Conditional Random Fields	25.2 %
Finite State Transducers	29.5 %
Support Vector Machines	29.6 %

4. Probabilistic logic and inference for slu

In practical applications, SLU is part of a dialogue system whose objective is the execution of actions to satisfy a user goal. Actions can be executed only if some pre-conditions are asserted true and their results are represented by post-conditions. Preconditions for actions can be formulated is in formal logic. Preconditions for actions depend on instances of semantic structures.

The system knowledge is made of general knowledge, e.g. knowledge about dates and time, and specific domain knowledge, e.g. the details of a telephone service. Let us call the resulting knowledge *in-domain knowledge*.

As a dialogue progresses, part of the domain knowledge is instantiated. The purpose of the dialogue is to interpret the user beliefs and goals and represent them with the MRL. Eventually, system actions like accessing a data base, are performed to satisfy a user request. If MRL contains frames, then user sentences should cause the instantiation of some frames, the assignment of values to some frame roles and functions to describe them. Instantiation is based on what the user says, but also on what can be inferred about the implicit meaning of each sentence.

Control strategies for interpretation determine how semantic structures are built, how expectations are defined and how knowledge structures are matched with input data in the presence of constraints and imprecision.

There are two basic types of strategy. One is based on path extraction from a semantic or a frame network. The other adopts a constructionist approach that can use one or more of the following methods: inference, parsing, abduction, agenda-based formation and scoring of interpretation hypotheses called *theories*.

In the constructionist approach, the meaning of a complex phrase is considered to be a function of the meanings of its constituent parts and the way in which these parts are syntactically combined. Reasoning is performed by programs that activate memory structures by placing activation markers on them. Nodes of the structure are activated when the corresponding concepts are instantiated. Active nodes may spread activation markers to hypothesize or predict the activation of concepts which have not yet been instantiated. When two markers collide in the same node, a path is identified indicating a possible inference. Frame-activated inference is discussed in (Norvig 1987).

Early approaches to SLU used semantic representations in terms of partitioned semantic networks (Walker, 1975). Marker propagation was used for making predictions about concepts likely to appear in the natural language messages. Concept hypotheses were generated by templates matching word and partial parses (obtained with a best first parser) with semantic structures.

In the Hearsay II SLU architecture (Erman et al., 1980), a heterarchical architecture was used for applying rules for matching and inference. An agenda based control strategy selects a rule whose precondition matches the content of a blackboard. If matching is successful, then actions are performed which modify the content of the blackboard.

The weakness of these approaches was that they did not contain an effective method for evaluating the confidence of the generated hypotheses.

If instances of semantic constituents are structured into probabilistic frames, it is possible to have a probability model for the values that can fill a slot (Koller, 1998). It is also possible to inherit probability models from classes to subclasses, to use probability models in multiple instances and to have probability distributions representing structural uncertainty about a set of entities.

It is shown that it possible to construct a Bayesian Network (BN) for a specific instance-based query and then perform standard BN inference if the graph obtained from a list of statistical dependencies between slot values is acyclic. Otherwise, Markov Logic Networks (MLNs) can be used (Richardson, 2006).

Probabilities obtained with these models can be combined with probabilities computed by SLU components in a way that is now introduced.

Let us consider an instance $\Gamma_{i,j}$ of a frame F_i.
Let us indicate by $\Gamma_{i,j} : \lfloor \gamma_{i,j,1}, \ldots\ldots, \gamma_{i,j,k}, \ldots\ldots\ldots, \gamma_{i,j,K} \rfloor$ the set of roles (slots) of $\Gamma_{i,j}$ that are instantiated and possibly filled by a value.

The instantiation of each slot is based on a casual relation graphically represented as follows:

$$Y_k \rightarrow W_k \rightarrow C_k \rightarrow \gamma_{i,j,k}$$

Y_k is a sequence of acoustic feature vectors from which a word sequence W_k has been hypothersized. W_k contains the support for a semantic constituent C_k expressed by a predicate in first-order logic. If C_k has been expressed in relation to $\Gamma_{i,j}$, then it becomes the slot hypothesis $\gamma_{i,j,k}$.

There may be other dependences between slot values, represented by lings in the following feature:

Each slot hypothesis can be evaluated by the following probability:

$$P\left[W_k, C_k, \gamma_{i,j,k} \middle| Y_k\right] \approx P\left[C_k, \gamma_{i,j,k} \middle| W_k\right] P\left[W_k \middle| Y_k\right] \approx$$

$$\approx \frac{P\left[W_k \middle| C_k, R(\gamma_{i,j,k})\right]}{P\left[W_k\right]} P(\gamma_{i,j,k}) P\left[W_k \middle| Y_k\right]$$

since $P\left[C_k \middle| \gamma_{i,j,k}\right] = 1$

The ratio $\frac{P\left[W_k \middle| C_k, R(\gamma_{i,j,k})\right]}{P\left[W_k\right]}$ can be obtained with two different language models (LMs) a generic LM for the denominator and an LM estimated on dialog turns expressing C_k and a relation $R(\gamma_{i,j,k})$ to an instance of F_i. Notice that C_k is hypothesized in a dialog turn using a specific concept LM and $P\left[W_k \middle| C_k, R(\gamma_{i,j,k})\right]$ could also be approximated by $P\left[W_k \middle| C_k)\right]$ obtained directly with this concept LM. Notice also that if the LMs are estimated on entire turns rather than concept supports, the ratio of probabilities will be mostly determined by the n-grams of the words characterizing the supports, especially if unigram LMs are considered. The LM used for computing the numerator can also be obtained by interpolating a generic LM with a relation specific one.

As an evidence indicator for the entire instantiation $\Gamma_{i,j}$, let us define the following vectors
$C_{i,j} : [C_1, \ldots\ldots, C_k, \ldots\ldots\ldots, C_K]$,
$W_{i,j} : [W_1, \ldots\ldots, W_k, \ldots\ldots\ldots, W_K]$,
$Y_{i,j} : [Y_1, \ldots\ldots, Y_k, \ldots\ldots\ldots, Y_K]$. Assuming also that each concept has a support that is somehow different from the supports of other concepts and assuming independence among supports, one gets:

$$P\left\{\Gamma_{i,j}, C_{i,j}, W_{i,j} \middle| Y_{i,j}\right\} =$$

$$= P\left[\Gamma_{i,j}\right] \left\{ \prod_{k=1}^{K} \frac{P\left[W_k \middle| C_k, R(\gamma_{i,j,k})\right]}{P(W_k)} \right\} P\left[W_{i,j} \middle| Y_{i,j}\right]$$

Confidence indicators can be introduced to replace some probabilities. Let $\Phi_{i,j} = \left\lfloor \varphi_{i,j,1}, \quad , \varphi_{i,j,k}, \ldots\ldots, \varphi_{i,j,K} \right\rfloor$ be a vector of confidence indicators, one for each slot. In this case, the following computation can be performed:

$$P\left\{\Gamma_{i,j} \middle| \Phi_{i,j}\right\} = \frac{P\left\{\Phi_{i,j} \middle| \Gamma_{i,j}\right\} P\left[\Gamma_{i,j}\right]}{P\left\{\Phi_{i,j}\right\}}$$

Vector quantization can be introduced for $\Phi_{i,j} = \left\lfloor \varphi_{i,j,1}, \quad , \varphi_{i,j,k}, \ldots\ldots, \varphi_{i,j,K} \right\rfloor$.

User goals can be represented by frames. A plan for achieving each goal can be represented by a sequence of states. If different goals are hypothesized in a dialog control agenda, then the set of the corresponding plans are represented by a finite state machine. This corresponds to represent by a state a cluster of instances $\Gamma_{i,j}, C_{i,j}, W_{i,j}$ corresponding to successive slot filling of a frame instance.

As different states can be reached with different probabilities, a set of states can be active at a turn k of a dialogue A system was proposed in (Damnati, 2007) which interprets a dialogue turn message in two phases. In the first phase, a word-to-constituent transducer translates a word lattice into a constituent lattice. In the second phase, a set of precondition-action rules encoded as a transducer transforms concept hypotheses into state transitions. A lattice of words is thus translated into a set of states with attached probabilities p(S|Y) where S is a dialogue state and Y is the acoustic description of a spoken message.

The results reported in Table 3 are obtained with system 3000 data, using this approach (strategy 2) and are compared with the results obtained with a pure sequential solution (strategy1) consisting in taking the 1-best word sequence and mapping it into the 1-best concept sequence. The abbreviations are defined in tables 1 and 2, WER stays for Word Error rate.

Table 3 – Performance on goal detection using a two different strategies

	WER	IER
strategy 1	40.1	15.0
strategy 2	38.2	14.5

5. Conclusions

A modular SLU architecture has been introduced. It uses CRFs, classifiers and stochastic FSMs, which are approximations of more complex grammars, for generating semantic constituent hypotheses and probabilistic logic for performing semantic compositions.
Annotating corpora for these tasks is time consuming suggesting that it is suitable to use a combination of knowledge acquired by a machine learning procedure and human knowledge (Riccardi, 2005). Finding the best combination of these approaches is still a research issue. Other challenging problems concern the use of probabilistic logic, the introduction of suitable confidence indicators (as in Sarikaya, 2005), the design of interpretation strategies and their integration with dialog management.

6. Acknowledgements

The work described in this paper is performed in the European project LUNA, IST contract no 33549. (www.ist-luna.eu). Warm thanks to F. Béchet, G. Damnati, F. Duvert, M..J. Meurs, C. Raymond, C. Servan (LIA Avignon) and H. Ney, S. Hann (RWTH, Aachen).

7. References

G. Damnati, F. Bechet, R. de Mori, (2007) Spoken language understanding strategies on the France Telecom 3000 voice agency corpus, IEEE International Conference on Acoustics, Speech and Signal Processing, Honolulu, Hawai

R. De Mori, *Spoken dialogues with computers* Academic Press, 1998.

E.W. Drenth and B.Ruber (1997) Context-dependent probability adaptation in speech understanding *Computer Speech and Language*, 11(3):225-252.

H. Erdogan, , R., Sarikaya, S.F Chen, Y.Gao and M Picheny (2005) Using Semantic Analysis to improve Speech Understanding Performance. *Computer Speech and Language*, 19(3):321-344.

L. D. Erman, F. Hayes-Roth, V. R. Lesser et R. D. Reddy. The Hearsay-II Speech Understanding System : Integrating Knowledge to Resolve Uncertainty. *ACM Computing Surveys*, 12(2):213-253, 1980.

C. J. Fillmore (1968). The case for case. in E. Bach and R. Harms eds. *Universals in linguistic theory*, Holt, Rinehart and Winston, New York, 1968.

A. L. Gorin, G. Riccardi, and J. H. Wright, 1997 How may I help you? *Speech Commununication* vol. 23, no. 1–2, pp. 113–127, October.

D. Hakkani-Tur, F. Bechet, G. Riccardi and Gokhan Tur (2006) Beyond ASR 1-Best: Using Word Confusion Networks for Spoken Language Understanding, *Computer Speech and Language* 20(4):495-514.

R.Jackendoff (1990). *Semantic Structures*, The MIT Press, Cambridge Mass.

R.Jackendoff (2002). *Foundations of language*, Oxford University Press, Oxford UK.

T. Kawahara, K. Tanaka and S. Doshira (1999) Virtual fitting room with spoken dialogue interface. *Proc. ESCA Workshop on Interactive Dialog in Multi-Modal Systems,* Kloster Irsee Germany, pp.5-8, 1999

M. Kifer, G. Lausen and J. Wu (1995) Logical Foundations of Object-Oriented and Frame-Based Languages *Journal of the Assocmtmn for Computmg Machlncry*. 42 (4), July1995. pp. 741-843

D. Koller and A. Pfeffer (1998) Probabilistic frame-based systems. *Proc. AAAI98*, pages 580–587, Madison, Wisc.,

J. McCarty and P.J. Hayes. (1969). Some philosophical problems from the standpoint of artificial intelligence. *Machine Intelligence*, Ed. by B. Meltzer and D. Michie, Edimburg University Press.

M. McTear (2006) Spoken language understanding for conversational dialog systems, IEEE/ACL Workshop on Spoken Language Technology Aruba, December 2006

B. Minescu, G Damnati, F. Béchet, R. De Mori (2007) Conditional use of Word Lattices, Confusion Networks and 1-best string hypotheses in a Sequential Interpretation Strategy Proc. European Conference on Speech Communication and Technology, Interspeech 07, Antwerpen, Belgium

A. Nasr, Y. Estéve, F. Béchet, T. Spriet, R. de Mori (1999) A language model combining n-grams and stochastic finite state automata. Proc. Eurospeech-99, Budapest, Hungary, pp :2175-2178

N. Nilsson., (1981) *Principles of Artificial Intelligence* Tioga Press, 1981

P. Norvig (1987) Inference in Text Understanding. Proceedings Sixth National Conference on Artificial Intelligence, July 13-17 Seattle, WA. AAAI Press 561-565

F. Pereira (1990) Finite-state approximations of grammars Proc. DARPA, Speech and Natural language workshop, Hidden Valley, PA, June 1990, pp. 12-19
http://dblp.uni-trier.de/rec/bibtex/conf/aaai/Norvig87
R. Pieraccini, E. Levin and C.H. Lee (1991). Stochastic Representation of Conceptual Structure in the ATIS Task. Proceedings of the, 1991 *Speech and Natural Language Workshop*, 121-124, Morgan Kaufmann publ, Los Altos, CA.

S.S. Pradhan, W. Ward, . K. Hacioglu, J.H. Martin and D. Jurafsky, .(2004) Shallow Semantic Parsing using Support Vector Machines Proc HLT-NAACL Conference, Boston, Massachusetts, USA, May 04, pp.233-240.

N. Prieto, E. Sanchis and L. Palmero (1994) Continuous speech understanding based on automatic learning of acoustic and semantic models. Proc ICSLP 1994, Yokohama

O. Rambow, S. Bangalore, T. Butt, A. Nasr, R. Sproat (2002) Creating a Finite State Parser with Application Semantics , 19th International Conference on Computational Linguistics (Coling 2002) Taipei

C. Raymond, F. Béchet R. de Mori and G. Damnati, (2006) On the use of finite state transducers for semantic interpretation, *Speech Communication*, 48(3): 288-304, March 2006.

G. Riccardi and A.L. Gorin (2000) Stochastic language adaptation over time and state in natural spoken dialog *systems IEEE Transactions on Speech and Audio Processing,* SAP-8(1):3-10.

G. Riccardi and D. Hakkani-Tur (2005) Active Learning: Theory and Applications to Automatic Speech Recognition *IEEE Transactions on Speech and Audio Processing*,SAP-13 (4) : 534-545

M. Richardson and P. Domingos (2006) Markov Logic Networks, *Machine Learning*, 62:107-136.

R. Sarikaya, Y., Gao, M. Picheny and H. Erdogan (2005) Semantic Confidence Measurement for Spoken Dialog Systems *IEEE Transactions on Speech and Audio Processing*,SAP-13 (4) : 534-545

D. Walker, (1975) The SRI speech understanding system *IEEE Transactions On Acoustics, Speech, And Signal Processing*, ASSP-23, NO- *5*, pp. 397-416

Ye-Yi Wang; Li Deng; A. Acero, “Spoken language understanding.” *IEEE Signal Processing Magazine.* 22 (5) pp. 16- 31, 2005.

W. A. Woods(1975). What’s in a link? in D.G. Bobrow and A. Collins Eds, *Representation and understanding* , Academic Press, New York.