Learning How to Understand Language

Roberto Pieraccini, Esther Levin

AT&T Bell Laboratories
600 Mountain Avenue
Murray Hill NJ 07974 USA

Enrique Vidal

Departamento de Sistemas Informáticos y Computación
Universidad Politeécnica de Valencia
Camino de vera s/n 46071 Valencia, Spain

Abstract

In this paper we propose a learning paradigm for the problem of understanding spoken language. The basis of the work is in a formalization of the understanding problem as a communication problem. This results in the definition of a stochastic model of the production of speech or text starting from the meaning of a sentence. The resulting understanding algori thm consists in a Viterbi maximization procedure, analogous to that commonly used for recognizing speech. The algorithm was implemented for building a module, called *conceptual decoder* for the decoding of the conceptual content of sentences in an airline information domain. The decoding module is the basis on which a complete prototypical understanding system was implemented and whose performance are discussed in the paper. The problems, the possible solutions and the future directions of the learning approach to language understanding are also discussed in this paper.

1. Introduction - The problem of understanding language

Generally the label *understanding system* is granted to those pieces of software that prove some abilities to successfully interact with a, more or less trained, human being using a, more or less restricted, subset of natural language. The most populated category of those systems, also called *natural language interfaces*, corresponds to machines that are designed to retrieve information when provided with a question in natural language. How to build a natural language interface is quite well understood [6] provided we have enough knowledge about the characteristics of the specific subset of the language used in the application. Those characteristics are specified through grammars that are build in general by hand and strongly depend on the application task. Unfortunately for any specific language we may consider there is not such a thing like a general grammar that covers all the possible applications. It is quite clear that even though we had a *general* grammar of a language, it needed a sort of adaptation for dealing with a particula task. The problem is even more serious when we deal with spontaneous speech rather than with written language. Spontaneous speech is often ungrammatical, and idiomatic. Besides, in spoken language, there are phenomena like false starts and broken sentences that do not appear in written language. The idea of building a machine that is able of learning how to understand is thus rather appealing, but cannot be implemented without a substancial training corpus of proper examples of sentences and dialogues. The availability of the corpus is as impotant as the theory for developing the learning strategy. The evidence of this statement is supported by the recent history of ... this method in handling large vocabulary speaker independent continuous speech. In fact when the 1000-word Resource Management database was designed and made available within a DARPA project [9], the algorithms previously developed by many researchers were compared and refined in order to get maximum performance on a common task, although an undoubted portion of the success can be ascribed also to the huge increase in the power of the computers, allowing now to develop training algorithms able to estimate models from hours and hours of speech. However, while it is easy (although not trivial) to agree on which is the amount of information necessary to include in a speech corpus in order to be able to use it for learning acoustic models (generally the trascription of sentences into words carries enough information for estimating phonetic HMMs), when semantics is taken into account it is very hard to find a common suitable representation. This is because the kind of representation used for the meaning of a sentence depends generally on the particular understanding system that is being developed. However, the DARPA ATIS corpus [15] that was designed for the development of speech understanding systems is a good set of data for developing and testing some of the learning theories, and although it does not contain any explicit representation of the meaning of sentences, it contains other useful related informations that can be used within the framework of a learning strategy.

2. Understanding as a translation process

A natural language understanding system is a machine that produces an action as the result of an input sentence (speech or text). Recently, several researchers proposed to look at the understanding process as to a translation (or transduction) process (see Fig. 1) composed of two functional blocks. The first, called *semantic translator,* analyzes the input sentence in natural language *(N-L)* and generates a representation of its meaning in a formal semantic language *(S-L)*. The *action transducer* converts the meaning representation into statements of a given computer language *(C-L)* for executing the required action.
While the input natural language is given, and very little can be done for adapting it to our system except imposing constraints, we have the choice of designing the formal semantic language in order to make easier the task of building both the semantic translator and the action transducer. However, the boundary between the first and the second module is quite arbitrary. In [30], for instance, an automatic system is designed for translating English sentences directly into SQL queries, and in [16] there is an example of a system that goes from an English sentence ...

we may find that learning the parameters of the *semantic translator* becomes quite a difficult problem when the application entails a rather complex semantics.

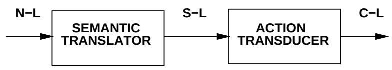

Figure 1: Understanding as a translation process

The choice of the semantic language strongly influences the performance of the understanding. For instance, it must be powerful enough to allow the representation of the semantics of the possible sentences in the application, even in presence of complex linguistic phenomena like relative and embedded clauses, and it should allow the implementation of a mechanism for the resolution of anaphoric and elliptical references. A semantic language with these characteristics must have, at least, the representation power of a context free grammar, thus allowing to represent the meaning of a sentence in the form of a tree [2] or of a network [5]. However, the complexity of the algorithms required for dealing with context free grammars, like for instance the *inside-outside* algorithm [4, 25] and the lack of training data for a specific language understanding application tend to convince the reasearchers to use a simpler approach. Moreover, if one analyzes the individual sentences that a subject asks in a conversation whose goal is to get some information from a database, very rarely one finds complicated constructs and subordinated clauses. Rather there are strong ungrammaticalities and hesitations, but quite often the sentences can be easily segmented into phrases, each one of them specifying a distinct concept. This is the observation that led to several prototypical learning systems [19, 22, 21]. The main assumption on which those systems are based can be expressed in the following terms. The meaning of a given sentence can be expressed by a *sequence of meaning units*, and this sequence can be put in sequential correspondence with portions (phrases) of the sentence. Of course this is a very strong assumption and it does not take an expert linguist for finding lots of examples in any language that violate this assumption. However, in limited but realistically defined semantic contexts this assumption holds for the majority of sentences. An example of this kind of representation is shown in Table 1. In the shown examples the meaning of a sentence is represented by a sequence of *keyword/value* pairs $m_j = (k_j, v_j)$, where $k_j \in \Gamma = \{\gamma_j\}$, is a conceptual category (i.e. a *concept* like for instance *origin of a flight, destination, meal*) and v_j is the *value* with which k_j is instantiated in the actual sentence (e.g. *Boston, San Francisco, breakfast*).

2..1 ATIS: an Example of a Speech Undertstanding Corpus

A relatively large corpus expressively designed for speech understanding (but not for learning) is being developed within the DARPA ATIS project [15]. ATIS stands for Air Travel Information System and the task is built around a subset of the OAG (Official Airline Guide) database, including 10 American cities. A corpus of spontaneous sentences is being collected and annotated by different sites [23]. The corpus is collected through a Wizard of Oz paradigm. Each subject is given a scenario and a travel planning problem to solve. The subjects are requested to solve the problem by interacting with a machine (that is actually a human *wizard*). The partial and the final responses of the machine are presented to the subjects via a display or a speech synthesizer. The sentences uttered by the subjects are recorded, transcribed and annotated carefully. Although the ATIS corpus may not be the best corpus for testing a semantic learning paradigm, it is readily available and it includes some kind of meaning annotation that can be indirectly used for our purpose.

Assessing the performance of a language understanding system is still an open problem mainly because the concept of *correct answer* is generally ambiguous and must be based on defined conventions that are not task independent. The DARPA community agreed upon scoring answers, within the ATIS task, by comparison with given reference answers that are produced for each valid sentence of the corpus. Of course the problem of the definition of a *correct* answer still remains. For instance, for a question like

> *SHOW THE LATE EVENING FLIGHTS BETWEEN BOSTON AND DALLAS*

the correctness of the answer depends upon the conventional definition of *late evening.* Then, once a time interval has been defined for late evening, it is still not clear what is the information to be listed. It could be the airline and flight number of each flight, but it could also include the departure time, the arrival time, the fare, and so on. A special committee within the DARPA community agreed upon a certain number of rules, called *principles of interpretation* [23], that should rule the majority of cases. Besides, it was also agreed on using two reference answers, namely a *minimal* and a *maximal* reference answer. An answer is thus considered correct if it contains all the information included in the minimal reference answer and no more than the information included in the maximal reference answer.

3. Formalization of the Understanding Problem - the Stochastic Approach

The stochastic approach to language understanding (see [19, 27, 26, 28]) is based on the *noisy channel* paradygm that was introduced for formalizing the general speech recognition problem in [3] and that constitutes today the theoretical basis of most of the current working speech recognizers. A version of this paradigm was recently proposed for formalizing the problem of automatic translation between two languages [10]. We use the same

SHOW ME THE FLIGHTS TO BOSTON	(question,display) (subject,flight) (destin,BBOS)
HOW MUCH IS THE PRICE OF THE FLIGHT FROM ATLANTA	(question,display) (subject,fare) (destin,MATL)
IS BREAKFAST SERVED ON THE FLIGHT?	(question,yes-no) (subject,breakfast)

Table 1: Example of keyword/pair representations of simple phrases within the ATIS domain.

paradigm for formalizing the problem of speech/text understanding [19]. The first assumption we make is that the meaning of a sentence can be expressedby a sequence of basic units $\mathcal{M} = \mu_1, \mu_2, \ldots, \mu_{N_M}$ and that there is a *sequential correspondence* between each μ_j and a subsequence of the acoustic observation $\mathbf{A} = a_1, a_2 \ldots a_{N_A}$, so that we could actually segment the acoustic signal into consecutive portions, each one of them corresponding to a phrase that expresses a particular μ_i. The second assumption consists in thinking of the acoustic representation of an utterance as a version of the original sequence of meaning units corrupted by a noisy channel whose characteristics are generally unknown. Thus, the problem of understanding a sentence can be expressed in this terms: given that we observed a sequence of acoustic measurements $\mathbf{A}$ we want to find which semantic message $\mathcal{M}$ most likely produced it, namely the one for which the a posteriori probability $P(\mathcal{M} \mid \mathbf{A})$ is maximum. Hence the problem of understanding a sentence is reduced to that of maximum a posteriori probability decoding (MAP). This formulation of the understanding problem leads to the maximization of the product of three factors, namely:

$$\max_{\mathbf{W},\mathbf{C}} P(\mathbf{A} \mid \mathbf{W})P(\mathbf{W} \mid \mathbf{C})P(\mathbf{C}) \tag{1}$$

The first one, the acoustic model $P(\mathbf{A} \mid \mathbf{W})$, is the probability of a sequence of acoustic observations given a sequence of words. Models for the maximization of this probability are well undertstood, and are generally implemented in the form of acoustic Hidden Markov Models [3, 31]. The second term, the *syntactic model* $P(\mathbf{W} \mid \mathbf{C})$, is the probability of a sequence of words $\mathbf{W} = w_1, \ldots, w_{N_W}$ given a sequence of conceptual labels $\mathbf{C} = c_1, \ldots, c_{N_W}, \; c_i \in \Gamma$. And finally, the semantic term $P(\mathbf{C})$ expresses the probability of a sequence of conceptual labels. The syntactic and the semantic terms can be combined in a single model that, with some additional assumptions takes the form of a Hidden Markov Model whose states correspond to conceptual labels and whose obervations are sequences of words modeled, for each state, with a bigram language model. Formally the conceptual model is defined by a set of states $\Gamma = \{\gamma_1, \gamma_2, \ldots \gamma_N\}$, a set of *concept conditional bigrams* $P(w_i \mid w_{i-1}, \gamma_i)$ and the *concept transition probabilities* $P(\gamma_i \mid \gamma_{i-1})$. Then, given a sequence of words the sequence of conceptual labels can be recovered by means of Viterbi decoding.

3..1 Training the Conceptual Model

Once the concept dictionary has been designed, based on the knowledge of the task, the concept conditional bigrams and the concept transition probabilities can be estimated from a corpus of training examples. Each training example consists of a sentence and the sequence $\mathbf{C}$ of associated conceptual labels. Unfortunately the conceptual labels must be provided manually for each sentence. The cost of handlabeling all the sentences in a big corpus can be comparable and even greater than that of writing a grammar for the application language. If this is the case, there are no advantages in using a learning system rather than a traditional one. Thus it becomes important to develop strategies for reducing the cost of conceptual annotation of the sentences in a big corpus. The annotation cost can be reduced either devising a strategy that takes advantage of all the possible semantically related additional information that is already in the corpus, or by making the annotation very simple and performable by non specialized people.

For instance, in the ATIS corpus, the meaning of sentences is not available in a declarative form. Instead, each sentence is associated with the *action* resulting from the *interpretation* of the meaning, namely the correct answer. One way of using this information for avoiding the handlabeling and segmentation of all the sentences in the corpus consists in creating a training loop in which the provided correct answer serves the purpose of a feedback signal. In practice all the available sentences are analyzed by the understanding system obtained with an initial estimate of the conceptual model parameters. The answers are then compared to the reference answers and the sentences are divided into two classes. The *correct* sentences, for which we assume that the conceptual segmentation obtained with the current model is correct, and the *problem sentences*. Then the segmentation of the correct sentences is used for reestimating the model parameters, and the procedure is repeated again. The procedure can be repeated until it converges to a stable number of correct answers. Eventually, the remaining *problem sentences* are corrected by hand and included in the set of correct sentences for a final iteration of the training algorithm. This procedure proved effective for reducing the amount of handlabeling. In the experiment described in [27] we showed that the performance increase obtained with the described training loop, without any kind of supervision (the remaining *problem* sentences were excluded from the training corpus) is equivalent to that obtained with the supervised smoothing. This means that the training loop, although is not able to learn radically new expressions or new concepts, is able to reinforce the acquired knowledge and to *infer* the meaning of semantically equivalent words. Of course the training loop cannot solve completely the annotaion problems. We noticed that more than 20% of the sentences in a corpus must be manually inspected and conceptually annotated. However the annotation operation can be made very simple making the semantic language *S-L* simple and flexible. The methodology used for annotating teh ATIS corpus [23] constitutes a good example of this idea. In fact the annotators rephrase each valid sentence in an artificial language that is a very restricted form of English. This *pseudo-English* rephrasing (called *win* or *wizard input*) constitute the input of a parser, called NL-parse [12], that unambiguously generates the SQL query. For instance, for a sentence like:

> *I'D LIKE TO FIND THE CHEAPEST FLIGHT FROM WASHINGTON D C TO ATLANTA*

The *win* rephrasing is:

> *List cheapest one direction flights from Washington and to Atlanta*

and the corresponding associated SQL statement is:

```
(SELECT DISTINCT flight.flight_id FROM flight WHERE (flight.flight_id IN (SELECT flight_fare.flight_id FROM flight_fare WHERE flight_fare.fare_id IN (SELECT fare.fare_id FROM fare WHERE fare.one_direction_cost = (SELECT MIN ( fare . one_direction_cost ) FROM fare WHERE fare.fare_id IN (SELECT flight_fare.fare_id FROM flight_fare WHERE flight_fare.flight_id IN (SELECT flight.flight_id FROM flight WHERE (flight.from_airport IN (SELECT airport_service.airport_code FROM airport_service WHERE airport_service.city_code IN (SELECT city.city_code FROM city WHERE city.city_name = 'WASHINGTON' )) AND flight.to_airport IN (SELECT airport_service.airport_code FROM airport_service WHERE airport_service.city_code IN (SELECT city.city_code FROM city WHERE city.city_name = 'ATLANTA' )))))))) AND (flight.from_airport IN (SELECT airport_service.airport_code FROM airport_service WHERE airport_service.city_code IN (SELECT city.city_code FROM city WHERE city.city_name = 'WASHINGTON' )) AND flight.to_airport IN (SELECT airport_service.airport_code FROM airport_service WHERE airport_service.city_code IN (SELECT city.city_code FROM city WHERE city.city_name = 'ATLANTA' )))));.
```

Both the SQL query and the *win* sentence can be considered semantic representations of the original sentence. In fact the SQL query is the final target of the understanding system and can be unequivocally obtained from the *win* sentence through an existing parser. Obviously the sequential correspondence assumption is strongly violated for the SQL representation. However a sequential correspondence can be easily found between the pseudo-English *win* sentence and the original message, at least for the shown examples. Since all the valid sentences in the ATIS corpus have a *win* annotation, the pseudo-English language can be thought of as an alternate candidate for the meaning representation in our learning framework. Using *win* for representing the meaning may lead to two different solutions. In the first we can think of developing a system that learns how to translate natural language sentences into pseudo-English sentences and then use the existing parser for generating the SQL query. In the second solution each *win* sentence in the corpus can be translated in the corresponding conceptual representation used for CHRONUS. This translation is unambiguous (*win* is an unambiguous artificial language by definition). A parser can be easily designed for performing the translation or, simply use CHRONUS itself for performing the translation[1].

3..2 The Sequential Correspondence Assumption

In section 3. we based our formalization of the speech understanding problem on the assumption that there is a sequential correspondence between the representation of a sentence (words or acoustic measurements) and the corresponding representation of meaning. Unfortunately, when using a simple annotation procedure like the *win* language, the sequential correspondence assumption will not hold for a good percentage of the sentences. A typical example is constituted by the following sentence:

> *COULD YOU PLEASE GIVE ME INFORMATION CONCERNING AMERICAN AIRLINES A FLIGHT FROM WASHINGTON D C TO PHILADELPHIA THE EARLIEST ONE IN THE MORNING AS POSSIBLE*

whose corresponding *win* annotation is:

> *List earliest morning flights from Washington and to Philadelphia and American.*

The problem of reordering the words of *win* representation for aligning it with the original sentence is a complex problem that cannot be solved optimally. Suboptimal solutions with satisfactory perfomance can be developed based on effective heuristics. We will not discuss the details of how the reordering can be put into practice. Rather we want to emphasize the fact that an iterative algorithm based on a model similar to that explained in section 3. led to almost 91% correct alignments between English sentences and corresponding *win* representations on a corpus of 2863 sentences. With additional refinements this technique can be used, integrated in the training loop, for automatically processing the training corpus of the conceptual model.

3..3 Interfacing with a Speech Recognizer

The most natural way of interfacing the conceptual decoder with a speech recognizer is by implementing the maximization of equation 1. This requires to implement a decoder that explores a network obtained by explicitly instantiating acoustic HMMs [13] representing words of the vocabulary for any concept. For a task like ATIS the dimension of the resulting network can be rather large. In theory, if there are 50 conceptual states and about 1,000 words, each one of them represented by a HMM with an average number of 15 states, the overall network is bound by a total number of 750,000 acoustic HMM states, with a number of connections of the order of 50,000,000

[1] *win* is a subset of natural English. Only a little adaptation was needed for developing a *win* translator based on the existing CHRONUS

(each conceptual conditional bigram model is represented by 1000×1000 connections). Of course not all the bigrams are observed or even possible in each state. If only those words and bigrams that were observed during the training are represented in a conceptual state, a more reasonable model can be obtained. In an experimental version of CHRONUS we estimated an integrated model with a total of 2400 HMM word models (corresponding to about 36,000 HMM acoustic states) and nearly 46,000 connections. This size of the model can be easily managed by a beam search recognizer [14]. The problem in using such a model is that while it constitute a reasonably good model for decoding the semantic message of a sentence, the limited amount of training data used for its estimation makes it a quite coarse model for constraining the speech recognition process. When bigrams of words that were not observed in the training data are actually uttered, the recognizer is forced to substitute them for known bigrams. Hence the recognition errors are propagated along the sentence, resulting in relatively poor recognition performance.

Smoothing techniques can be applied for estimating the probability of unobserved bigrams, like for instance methods relying on the Good-Turing estimation of probabilities [7]. This will increase the complexity of the model by allowing all the possible bigrams in each state. However a factorization of the maximization of equation 1 [20] can still lead to reasonably good results at an acceptable complexity. Hence several solutions could be implemented, like best first coupling (the best first recognized sentence is given to the conceptual decoding), N-best coupling [17] and word lattice coupling [11].

4. Discussion and Conclusions

In this paper we propose a new paradigm for language understanding based on a stochastic representation of semantic entities called concepts. An interesting way of looking at the language understanding paradigm is in term of a language translation system. The first block in Fig. 1 translates a sentence in natural language (*N-L*) into a sentence expressed in a particular semantic language (*S-L*). The natural language characteristics are generally unknown, while the semantic language designed to cover the semantic of the application is completely known and described by a formal grammar. The second step consists in the translation of the sentence in *S-L* into computer language code *C-L* for performing the requested action. This second module can be generally (but not necessarily) designed to cover all the possible sentences in *S-L*, since both *S-L* and *C-L* are known. However, the boundary between the first and the second module is quite arbitrary. The subject of this paper deal with the investigation of the possibility of automatizing the design of the first block (i.e. the *semantic translator*) starting from a set of examples. The semantic language chosen for the experiments reported in this paper is very simple and consists of sequences of keyword/value pairs (or *tokens*). There is no syntactic structure in the semantic language we use. Two sentences for which the difference in the semantic representation is only in the order of the tokens are considered equivalent. In this way we cover a good percentage of sentences in the domain, but still there are sentences that would require a structured semantic language. For instance the two following sentences are indistinguishable when represented by our semantic language, and obviously they have a different meaning.

> *IS THE EARLIEST FLIGHT GOING TO BOSTON ON A SEVEN FOUR SEVEN*

> *IS THE EARLIEST FLIGHT ON A SEVEN FOUR SEVEN GOING TO BOSTON*

The representation of this kind of sentences requires a more sophisticated semantic language that allows the use of bracketing for delimiting the scope of modifiers.

Although the system we propose uses a very simple intermediate semantic representation, it can successfully handle most of the sentences in a database query application like the ATIS task. When this simple representation is used and when the problem of *semantic translation* is formalized as a communication problem, a MAP criterion can be established for decoding the *units of meaning* from text or speech. The resulting decoder can then be integrated with other modules for building a speech/text understanding system.

An understanding system based on a learning paradigm, like the one proposed in this paper, can evolve according to different dimensions of the problem. One dimension goes with the increase in complexity of the semantic language *S-L*. Rather than using a sequential representation on could think of a tree representation of the meaning. However, this poses additional problems both in the training and decoding stage, and requires the use of algorithms designed for context-free grammars, like for instance the *inside-outside* algorithm [4][25] that have a higher complexity that those explained in this paper. Another dimension of the problem goes toward a complete automatization of the system, also for those modules that, at the moment, require a manual compilation of some of the knowledge sources. One of these modules is the *template generator*. Both [22] and [30] report examples of systems where the decision about the actual values of the conceptual entities (or an equivalent information) is drawn on the basis of knowledge acquired automatically from the examples in the training corpus. The kind of annotation required for the training corpus is also another dimension along with the research on learning to understand language should move. A strategy for learning the understanding function of a natural language system becomes really effective and competitive to the current non-learning methods when the amount of labor required for annotating the sentences in a training corpus is comparable or inferior to the amount of work required for writing a grammar in a traditional system. This requires the development of a learning system the does not require any other information than the representation of the meaning associated to each sentence (e.g. it does not require an initial segmentation into conceptual units, like in CHRONUS, for bootstrapping the conceptual models). Moreover, the representation of the meaning should be made using a *pseudo-natural* language, for making easier and less time consuming the work of the annotators. An example of this kind of annotation was introduced in section 3..2 with the pseudo-English *win* rephrasing. This suggests a possible evolution of the learning strategy for understanding systems toward a system starting with the limited amount of knowledge required for understanding a small subset of the whole language (e.g. the *win* language). Then the system can evolve to understanding larger subsets of the language using the language already acquired for rephrasing new and more complex examples. But, of course, the science of learning to understand is still in its infancy, and many more basic problems must be solved before it becomes an established solution to the design of a language interface.

References

[1] Baker, J., K., "Stochastic Modeling for Automatic Speech Understanding," in *Speech Recognition,* pp.521-541, Academic Press, New York, N.Y., 1975.

[2] Minski, M., "A Framework for Representing Knowledge," in P. Winston (ed.). *The Psychology of Computer Vision,* New York: McGraw-Hill, 211-277, 1975.

[3] Jelinek, F., "Continuous Speech Recognition by Statistical Methods," *Proceedings of IEEE,* vol. 64, no. 4, pp. 532-556, 1976.

[4] Baker, J., "Trainable Grammars for Speech Recognition," in Wolf, J. J., Klatt, D. H., editors, *Speech communication papers presented at the 97th Meeting of the Acoustical Society of America,* MIT, Cambridge, MA, June 1979.

[5] Sowa, J., *Conceptual Structures.* Reading, MA:Addison-Wesley, 1984.

[6] Allen, J, "Natural Language Understanding," The Benjamin/Cummings Publishing Company, Inc., 1987.

[7] Katz, S. M., "Estimation of Probabilities from Sparse Data for the Language Model Component of a Speech Recognizer," *IEEE Trans. on Acoustic, Speech and Signal Processing,* Vol. ASSP-35, No. 3, March 1987.

[8] Fissore, L., Laface, P., Micca, G., Pieraccini, R., "Lexical Access to Large Vocabularies for Speech Recognition," *IEEE Trans. on Acoustics, Speech and Signal Processing,* Vol. 37, No. 8, August 1989.

[9] Price, P., Fisher, W. M., Bernstein, J., Pallett, D. S., "The DARPA 1000-Word Resource Management Database for Continuous Speech Recognition," *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing,* ICASSP-88, New York, NY, April 1988.

[10] Brown, P. F, Cocke, J., Della Pietra, S. A., Della Pietra, V. J., Jelinek, F., Lafferty, J., Mercer, R. L., Roosin, P. S., " A Statistical Approach to Machine Translation," *Computational Linguistics,* Volume 16, Number 2, June 1990.

[11] Fissore, L., Kaltenmeier, A., Laface, P., Micca, G., Pieraccini, R., "The Recognition Algorithms," in *Advanced Algorithms and Architectures for Speech Understanding,* G. Pirani editor, Springer-Verlag Brussels -Luxemboug, 1990.

[12] Hemphill, C. T., Godfrey, J. J., Doddington, G. R., "The ATIS Spoken Language Systems, pilot Corpus," *Proc. of 3rd DARPA Workshop on Speech and Natural Language*, pp. 102-108, Hidden Valley (PA), June 1990.

[13] Lee, C.-H, Rabiner, L. R., Pieraccini, R., Wilpon, J. G., "Acoustic Modeling for Large Vocabulary Speech Recognition," *Computer Speech and Language*, **4** pp. 127-165, 1990.

[14] Pieraccini, R., Lee, C. H., Giachin, E., Rabiner, L. R., "An Efficient Structure for Continuous Speech Recognition," in *Speech Recognition and Understanding, Recent Advances, Trends and Applications,* P. Laface and R. De Mori editors, Springer-Verlag Berlin Heidelberg, 1992.

[15] Price, P. J., "Evaluation of Spoken Language Systems: the ATIS Domain," *Proc. of 3rd DARPA Workshop on Speech and Natural Language*, pp. 91-95, Hidden Valley (PA), June 1990.

[16] Gorin, A. L., Levinson, S. E., Gertner, A., "Adaptive Acquisition of Language," *Computer Speech and Language*, **5**, pp. 101-132, 1991.

[17] Ostendorf, M., Kannan, A., Austin, S., Kimball, O., Schwartz, R., Rohlicek, J. R., "Integration of Diverse Recognition Methodologies Through Reevaluation of N-Best Sentence Hypotheses," *Proc. of 4th DARPA Workshop on Speech and Natural Language*, Pacific Grove, (CA), February 1991.

[18] Pallett, D. S., "Session 2: Darpa Resource Management and ATIS Benchmark Test Poster Session," *Proc. of 4th DARPA Workshop on Speech and Natural Language*, Pacific Grove, (CA), February 1991.

[19] Pieraccini, R., Levin, E., Lee, C.-H., "Stochastic Representation of Conceptual Structure in the ATIS task," *Proc. of 4th DARPA Workshop on Speech and Natural Language*, Pacific Grove, (CA), February 1991.

[20] Pieraccini, R., Lee, C. H., "Factorization of Language Constraints in Speech Recognition," *Proc. 29th Annual Meeting of the Association for Computational Linguistics,* Berkeley, CA, June 1991.

[21] Prieto, N., Vidal, E., "Learning Language Models through the ECGI Method," *Proc. of EUROSPEECH 91,* Genova, Italy, September 1991.

[22] Giachin, E. P, "Automatic Training of Stochastic Finite-State Language Models for Speech Understanding", *Proc. of International Conference on Acoustics, Speech and Signal Processing*, ICASSP-92, San Francisco, CA, March 1992.

[23] *MADCOW*, "Multi-Site Data Collection for a Spoken Language Corpus,", *Proc. of Fifth Darpa Workshop on Speech and Natural Language*, Harriman, NY, Feb 1992.

[24] Pallett, D. S., Dahlgren, N. L., Fiscus, J. G., Fisher, W. M., Garofolo, J. S., Tjaden, B. C., "DARPA February 1992 ATIS Benchmark Test Results," *Proc. of Fifth Darpa Workshop on Speech and Natural Language*, Harriman, NY, Feb 1992.

[25] Pereira, F., Schabes, Y., "Inside-Outside Reestimation from Partially Bracketed Corpora," *Proc. of Fifth Darpa Workshop on Speech and Natural Language*, Harriman, NY, Feb 1992.

[26] Pieraccini, R., Tzoukermann, E., Gorelov, Z., Levin, E., Lee, C.-H, Gauvain, J.-L, "Progress Report on the Chronus System: ATIS Benchmark Results," *Proc. of Fifth Darpa Workshop on Speech and Natural Language*, Harriman, NY, Feb 1992.

[27] Pieraccini, R., Gorelov, Z., Levin. E., Tzoukermann, E., "Automatic Learning in Spoken Language Understanding," *Proc. of 1992 International Conference on Spoken Language Signal Processing*, ICSLP 92, Banff, Alberta, Canada, October 1992.

[28] Tzoukermann, E., Pieraccini, R., Gorelov, Z., "Natural Language Processing in the CHRONUS System," *Proc. of 1992 International Conference on Spoken Language Signal Processing*, ICSLP 92, Banff, Alberta, Canada, October 1992.

[29] Oncina, J., García, P., Vidal, E., "Learning Subsequential Transducers for Pattern Recognition Interpretation Tasks," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 15, No. 5, pp. 448-458, May 1993.

[30] Kuhn, R., De Mori, R., "Learning Speech Semantics with Keyword Classification Trees," *Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing,* ICASSP-93, Minneapolis, Minnesota, April 27-30, 1993,

[31] Rabiner, L. R., Juang, B. H., "Fundamentals of Speech Recognition," Prentice Hall Signal Processing Series, 1993.