On the Use of Structures in Language Models for Dialogue

Specific Solutions For Specific Problems

Yannick Estève, Christian Raymond, Renato De Mori
LIA-CNRS - University of Avignon – France

Abstract: Availability of large corpora for training language models to develop dialogue systems is rare. Fortunately, for specific dialogue application, many sentences follow a limited number of typical patterns. In a language like French, frequent errors are due to homophones.Three paradigms are proposed in this paper to rescore a trellis of hypothesized words. They are based on sentence patterns detected in the most likely sentence hypothesized in a first recognition phase.

Key words: language model, adaptation, stochastic finite state automata, rescoring, semantic classification trees, speech processing, dialogue

1. INTRODUCTION

Very often dialogue systems are developed without the availability of large corpora for training language models (LM). In spite of this, many sentences follow a limited number of typical patterns. Furthermore, many errors are due to minimal acoustic variations and correspond to sentences that are syntactically acceptable. Other errors are due to homophones, very frequent in a language like French.

This motivates the approach proposed in this paper which suggests to rescore a trellis of hypothesized words based on different types of LMs obtained by adapting to this specific problem some learning methods developed in Artificial Intelligence and Pattern Recognition. These methods are inspired by paradigms known as learning by analogy, explanation-based learning, error correcting parsing and semantic classification. The proposed

approach to LM structure learning is inspired by basic principles of the above mentioned methods and proposes new and effective solutions for dialogue applications.

Three models for hypotheses rescoring based on LMs will be introduced.

Generation of plausible trigrams by analogy consists in constructing new trigrams not observed in the training set, by replacing words or histories in an observed trigram which have analogy with words in the trigram and have with them a small distance. Distances between words and histories are measured in a reduced space obtained with Singular Value Decomposition (SVD) of the matrix of probabilities in which rows correspond to words and columns correspond to histories.

Generation of explanation-based error-correcting automata starts with the observation of an error, e.g. the absence of a verb, in the development and the formulation of the correct sequence of words. The error and its context are then generalized to obtain a precondition for automaton application and the correction is generalized leading to an automaton.

Homophone or quasi-homophone disambiguation using semantic classification trees (SCT) uses sentence patterns detected in the most likely sentence hypothesized in a first recognition phase. The case considered here is that of a confusions between very similar frequently used phrases, which may have similar or even the same phonetic transcription. An SCT generates a correction on a phrase of the hypothesized sentence based on the pattern which apply to the sentence. This is more than using automata, because corrections may involve a selected number of non contiguous words.

The application considered for the experiments is a prototype of vocal server developed at France-Telecom R&D. This is a medium size vocabulary (1000 words) telephone application involving requests of information about telephone services. Most of the recognition problems are related to the expression of a limited number of concepts characterized by a small set of keywords and a much larger number of specification words like geographic locations and job types.

As this is a real-world application, many recognition errors are due to background noise, hesitations, correction, erroneous end-point detection, use of out-of-vocabulary (OOV) words. Many errors lead to ungrammatical or

non-sense phrases and can be characterized by the absence of a verb or a noun, by incorrect articles or prepositions, by incorrect agreement.

Many errors appear on a fairly limited number of sentence types. Tangible reductions of Word Error Rates (WER) are obtained with different methods because frequent errors are of different type.

2. RELATED WORK

Although it is difficult to find precursors of the work described in this paper, it can be, to some extent, related to the attempt to use units in language modeling which are compounds of words.

In (Kaiser et *al.*, 1999) various methods for deriving finite-state machines for phrase structure grammars are discussed. A parser for word prediction as well as for extracting semantic interpretations is outlined.

Stochastic FSA are proposed in (Gotoh et *al.*, 1999) for Named Entity (NE) identification.

Another method for obtaining phrases is proposed in (Siu and Ostendorf, 2000b). Here the focus is on extending the context for particular words. This is then treated as a variable n-gram model even if, for computing the probability of a word given its history, the history is not extended for the whole distribution. The phrases are obtained by considering the variation in likelihood when the history of a word is extended.

A systematic approach to obtain phrases to be considered as units in a statistical LM is proposed in (Klakow, 1998). It is inspired from work in text compression, and uses the following criteria to join two words into a single compound:

- pair frequency
- mutual information
- change in the unigram likelihood of the entire corpus before and after the fusion

As full search is NP-complete, heuristics are introduced to make the computation feasible.

In (Kawahara and Doshita, 1999), a model for characterizing filler phrases depending on speaking style is proposed. The model should characterize topic-independent patterns that accompany key-words or key-

phrases. A topic-independent lexicon should be used even if it is important to notice that words common to different applications may be relevant for understanding. The LM is made of a key-phrase grammar with filler phrases obtained by concatenation of frequent filler words. Filler model results better than syllable filler model. This is particularly useful when sentences of a given topic have to be spotted in a general conversation.

In (Galescu and Ringer, 1999), syntactic and semantic processing are proposed to obtain *factoids*, i.e. word sequences like dates and times, and *captoids* (i.e. titles of books, movies etc.).

In (Souvignier and Kellner, 1998), word-graphs are rescored using n-grams as well as SCFG-based LMs. Adapted n-gram models are obtained by simply adding new counts for each item to the counts used for building the existing model. SCFG are used to extract language structures related to concepts, the rest being filled by n-gram fillers. Fillers and concept grammars are n-gram models.

In (Riccardi, 2000) it is proposed to generate word hypotheses based on the detection of morphemes which are entirely contained into the words. Morphemes are variable length phoneme sequences generated by VFSA and are automatically detected by finding the phoneme sequences for which the mutual information (computed with two phoneme sequences) is maximum.

In (Meteer and Rohlicek, 1993) a stochastic grammar for fragment of phrases is used. Its probabilities are combined with n-gram probabilities with a back-off smoothing scheme. This problem of combining stochastic grammars and n-grams is first discussed in (Mark et *al.*, 1992).

In (Wang, 2000) following (Nasr et *al.*, 1999) a sentence W is segmented into a sequence of segments each of which is either a word or a non terminal covering a sequence of words.

Two-level structures with regular expressions are proposed in combination with n-grams in (Galescu and Allen, 2000).

N-grams of semantic classes and grammars for each class are used for generating artificial data, not very natural, but with similar perplexity as correct ones (Fosler-Lussier and Kuo, 2001).

Dependency relations are used in relational head acceptors and transducers. Probabilistic generative models assign costs to acceptor actions (Alshavi et *al.*, 1997).

Structural, lexical punctuation and derivative cues are used for automatic detection of text genre (topic) (Kessler et *al.*, 1997).

In (Chelba and Jelinek, 1998) a *word-k prefix* of a sentence is considered. It is made of the first key words preceded by a zero-th sentence beginning symbol. To it a *word-parse k-prefix* is associated. It consists of a sequence of root-only trees plus binary subtrees. Each root-only tree is made of *exposed heads*, each being a pair of a word and a nonterminal symbol or a word and its pos tag. Binary subtrees are included into the parse tree and have a span which is completely included in the *word-k prefix.*

3. GENERATION OF PLAUSIBLE TRIGRAMS BY ANALOGY

If a limited amount of training data is available, many trigrams that would appear more than once in an ideally large training corpus have a probability computed with a back-off model. This probability is often much lower than the one that would be computed with a richer training corpus. Furthermore, in many practical applications, the training data available are biased by the fact that they have been collected with a limited number of speakers and in a limited time period. This has the effect that often the probability of certain trigrams is abnormally large.

These considerations suggest that trigram counts have to be adapted. The same adaptation algorithm can be applied to all trigrams or only to certain classes of them for which an algorithm can produce tangible benefits. For certain languages, like French, many errors are due to erroneous recognition of prepositions, articles and other short words appearing in a trigram which include a noun or a verb. For this reason, attention has been focused on trigrams involving the most frequent nouns (e.g. server, number, job, region) and verbs (e.g. call, find, look for, will), as well as geographic names and typical expressions like *toute la France* .

The number of trigrams considered is about 200, while the number of trigrams observed in a training set of 70,000 words is about 10,000. Each trigram t is represented as follow $t=hw$, where h represents the history of

word *w*. First of all, let us consider the generation of new trigrams sharing the same history. Let *t'=hv* be a new trigram derived from *t* by analogy. The possibility of acquiring new knowledge by analogy was first proposed by (Evans, 1968).

Derivation by analogy can be made by extracting from a very large general corpus all the trigrams *t''=hx* such that *x* belong to the same syntactic class of *w*. For a limited domain application and for a limited number of histories, generation by analogy can also be done manually. More formally, the generation by analogy of a set of trigrams T' given a set T of observed trigrams is defined as follows:

$$T' = \{t' = hx \mid [(t = hw) \in T] \wedge [POS(x) = POS(w)] \wedge [SEMCOMP(x,w)]\}$$

where POS(w) indicates the syntactic class (the Part Of Speech) of word *w*. SEMCOMP(x,w) indicates that x and w are semantically compatible words as it will be defined later on.

Let *c(hw)* be the count of trigram *t* obtained directly from the training set. Let:

$$c_M(h) = \max_{y \,/\, hy \in T} c(hy)$$

$$m(h) = \arg\max_{y \,/\, hy \in T} c(hy)$$

The new counts c'(hz) counts of trigrams having history *h* are recomputed as follows:

$$c'(hz) = \begin{cases} \beta_1 c(hz) & \text{if } c(hz) > \vartheta_1 c_M(h) \\ \alpha_1 c_M(h)\, e^{-d(x,m(h))} & \text{otherwise} \end{cases}$$

$\alpha_1, \beta_1, \vartheta_1$ are thresholds that can be settled in order to satisfy a condition on the sum of counts for history *h*. Threshold ϑ_1 is set in such a way that, if a count is more than 10% of the maximum count, it should just be multiplied by β_1. If very few of the possible trigrams, analogous because they have the same history, have very high counts, then it is likely that this is the result of a bias in the training set and part of their counts should be redistributed among the analogous trigrams.

For words z like prepositions or determiners, it is reasonable to assume that they have the same probability in phrases with history h, leading to the following assumption:

$$c'(hz) = \begin{cases} \beta_1 c(hz) & \text{if } c(hz) > \vartheta_1 c_M(h) \\ \alpha_2 c_M(h) & \text{otherwise} \end{cases}$$

Distance d(x,m(h)) is the Euclidian distance between each pair of word computed in (Janiszek et *al.*, 2001).

4. GENERATION OF EXPLANATION-BASED ERROR-CORRECTING AUTOMATA

Error correcting parser theory (Fu, 1982) uses knowledge to augment the rules of grammar so that a parser using the augmented grammar can parse erroneous sentences. A similar type of rules can be used to invoke regular languages accepted by finite-state automata to be dynamically associated to n-gram LMs in such a way that correct phrases become more likely than hypothesized phrases that are syntactically or semantically incorrect.

Detail of such a combination are described in (Nasr et *al.*, 1999) and have been extended to context-free languages in (Huang et *al*., 2001).

A method is introduced in the following for acquiring error correcting knowledge from examples. It is based on Explanation-Based Learning (EBL), a methodology for extracting general knowledge from specific examples (Mitchell *et al.*, 1986).

In our case, an example is a phrase *m* in *context (a,b)* which is incorrect and should be replaced by *n*. If the hypothesis contains the *context (a,b*), the phrase *x* between *a* and *b* is analyzed. A single observation is sufficient for detecting an inconsistence in *context (a,b)*. If it is inconsistent, an automaton *A(a,b)* can be derived manually or by grammatical inference (Fu, 1982) on all the phrases obtained from a general large corpus and having *context (a,b)* and words between *a* and *b* belonging to the application lexicon. Eventually, *context (a,b)* can also be generalized by considering synonyms of words in it. As the training set is limited, interesting cases can be found by simply running the recognizer on the training set.

The automaton *A(a,b)* is then combined with a n-gram model as explained in (Nasr et *al*., 1999). This new model is then used to rescore the trellis generated by the recognizer.

5. DISAMBIGUATION USING SEMANTIC CLASSIFICATION TREES

5.1 Semantic Classification Tree

A *Semantic Classification Tree* (SCT) is a binary tree with a question associated to each node. Each node has two successors, one is reached if the answer to the node question is YES, the other node is reached if the answer is NO. Questions are about sentence patterns made of words and wildcard symbols (+). If the node pattern matches with the sentence to be interpreted, then the answer to the node question is YES and the successor node pointed by the arc labeled YES is considered, otherwise the answer is NO and the corresponding successor node is considered.

The nature of the questions in the SCTs is such that the rules learnt are robust to grammatical and lexical errors in the input from the recognizer . In fact, these questions are generated in a manner that tends to minimize the number of words that must be correct for understanding to take place. Question generation involves ”gaps”: words and groups of words that can be ignored. Thus, each leaf of an SCT corresponds to a regular expression containing gaps, words, and syntactic units.

If one generalizes away from the domain-specific details, one can give the following recipe for building a SCT .

1. Collect a corpus of utterances in which each utterance is accompanied by its semantic representation.
2. Write a local parser that recognizes semantically important noun phrases that encode variables in the semantic representation (e.g., times, locations) and replaces such phrases with a generic code (while retaining a value for each variable).
3. Given a new utterance, one can generate a semantic representation from the resulting system as follows:
   - pass the utterance through the local parser,
   - temporarily strip out variable values and submit the resulting string to the SCT,

The most interesting aspect of SCTs is that the inference carried out explicitly models *don't care* words, allowing the system to tolerate a high degree of misrecognition in semantically unimportant words.

Experience gained in LM building for dialogue systems in the French language has shown problems that are probably common to many other

languages. Among them, it appears that a number of confusions arise between words that are phonetically very similar. We shall call these cases *quasi-homophones* including also the homophones which are fairly frequent in French. There may also be phrases including a few quasi homophones making the problem of disambiguating among them even more difficult.

Disambiguation between quasi-homophone candidates often requires knowledge which goes far beyond that contained in trigrams and is made of sentence patterns including POS, word classes, phrases and gaps. This is due to the fact that constraints for quasi-homophones are of semantic nature and can be automatically learnt with SCTs.

The proposed paradigm for quasi-homophone disambiguation is based on an automatic training method summarized as follows:

- use the training set to identify more frequent and confusable phrases,
- infer from the training set sentence patterns for these phrases,
- for each phrase *y* belonging to a type of phrase confusion, a classification tree is trained using a slightly modified version of the algorithm proposed in (Kuhn and De Mori, 1995) to infer sentence patterns for each acceptable sentence of that type.

An example of confusion is between phrases {*ce serveur, ces serveurs*} represented by the pattern of lems *pl= (ce* serveur*)*.

The classification tree shown in figure **1** was obtained from a training set of *9842* sentences. Detailed sub-trees at the bottom of the figure are represented by triangles for the sake of simplicity.

This classification tree is used with each hypothesis recognized on a first pass, when this hypothesis contains the phrase {*ce serveurs*} or {*ces serveurs*}.

Questions are based on the presence (or the absence) of particular words or parts-of-speech before or after the studied phrase. For example, in figure **1** the part-of-speech "XLOC", which represents names of countries or cities, is used. The words preceding the phrase {*ce serveur*} or {*ces serveurs*} are considered as the beginning of the sentence and are represented by *B*. Words following this phrase are representing as *E*.

Percentages of occurrences of phrases {*ce serveur*} and {*ces serveurs*} is associated to each node of the classification tree. These percentages will be used to rescore the hypothesis.

5.2 Computation of LM probabilities

Let W be a sentence hypothesis containing y, the sequence of words affected by pl. W can be seen as the concatenation of B (the sequence of words preceding y) y and E (the sequence of words following y). Let φ be the sentence context captured by the pattern selected by the classification tree. Sentence candidate rescoring is performed using the following LM probability:

$$P(W|\varphi) \cong P_g(B)\left\{P_b(y|BE)P(b|\varphi)+P_g(y|h)P(g|\varphi)\right\}\left\{P_g(E|yB)P(g|\varphi)+P_b(E|B)P(b|\varphi)\right\}$$

The subscript g indicates a general (in our case a trigram) model, the subscript b indicates the new model. In particular, the probability:

$$P_b(y|BE)$$

is computed from the leaves of the classification trees.

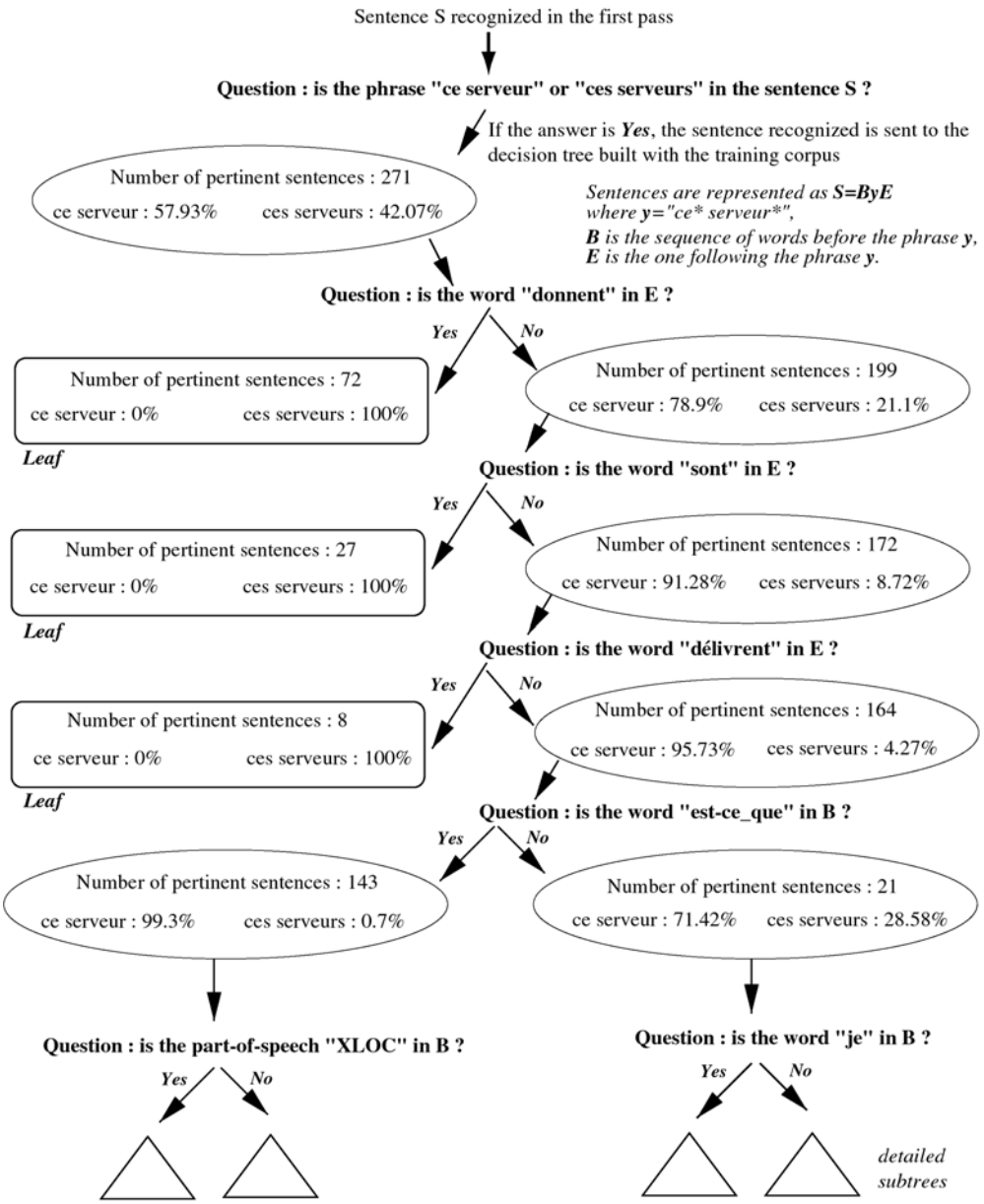

Figure 1. Example of Semantic Classification Tree

This probability can be used in various ways. In the experiments described in the following, it has been used to decide about corrections to be made on *s*. The correction which obtained the highest probability has been included between *B* and *E* to generate the recognition results.

In the probability $P_g(y|h)$, h is the history of y used by the general model.

6. EXPERIMENT RESULTS

A system, called AGS, described in (Sadek et *al.*, 1996), and deployed on the telephone network, performed a first step recognition for the test set, and made available, for each test sentence, a trellis of word hypotheses as well as the best hypothesis produced by the system. The purpose of the experiments is that of assessing if and how much the Word Error Rate (WER) can be reduced by rescoring the word hypotheses with new LMs by using the same scores provided by the acoustic models when the baseline hypothesis was generated.

Experiments were carried out using a telephone corpus of sentences from person-machine dialogues collected by France Telecom Research and Development in fairly severe conditions all over France. The training set contains *9842* sentences for a total of *49610* words. The vocabulary used within these experiments contains *878* words. Experiments were conducted by rescoring a test set of *1422* word-graphs. The Word Error rate using only trigrams is *21,8%*

Table 1 provides examples of the confusions for which a LM based on a Semantic Classification Tree was applied, the number of cases encountered in the test set, the number of correct recognition instances before and after the application of this new model.

Pattern pl	Number of cases	Correct in baseline	Correct with special LM
quels/quelles	31	18	27
numéro/numéros	42	37	39
ce/ces serveur(s)	38	32	38

Table 1. *Performance of the proposed method on specific confusion sets*

Experiments on the most frequent homophones and quasi homophones of the same type as those shown in Table 1 have lead to a WER reduction of 28.9% for the cases where these homophones appeared.

Results reported in Table 2 show the improvement introduced by the use of analogy, error correction and SCT-based model. As each paradigm is used for specific problems which rarely concern another one, their improvements are cumulated : a 15,5% total WER reduction is obtained.

LM	WER
Baseline	21.8 %
+ Analogy	20.1 %
+ error correcting automata	18.7 %
+SCT	18.4 %

Table 2. *Performance of the proposed method on specific confusion sets*

7. CONCLUSIONS

The paradigms suggested in this paper improve results for specific cases. Usually, works for language modelling to improve performances of speech recognition processing propose general approaches, whereas method presented here is only used on precise problems. In the future, more specific paradigms can be proposed : the sum of their small improvements should generate an interesting reduction of the Word Error Rate.

8. REFERENCES

H. Alshavi, A.L. Buchsbaum and F. Xia (1997). A comparison of head transducers and transfer for limited domain translation application. *Proc. of the ACL conference*, Madrid, Spain pp.360-365

C. Chelba and F. Jelinek (1998). Exploiting syntactic structure for language modelling. *Proc. of the COLING-ACL conference*, 1998, Montreal, QC, Canada., 225-231

T.G. Evans (1968). A program for the solution of a class of geometric analogy intelligence test questions. In M. Minsly Ed., *Semantic Information Processing,* 271-353. MIT Press. Cambridge MA.

Fosler-Lussier and H.K.J. Kuo (2001). Using semantic class information for rapid development of language models within ASR dialogue systems. *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, salt Lake City, UT

K.S. Fu (1982). Syntactic Pattern Recognition. *Theory and Applications*, Prentice Hall.

L. Galescu and E.K. Ringger (1999). Augmenting words linguistic information for n-gram language models. *Proc. of Eurospeech99*, Budapest, Hungary

Galescu and J. Allen (2000). Evaluating hierarchical hybrid statistical language models. *Proc. International Conference on Spoken Language Processing*,Beijin

Y. Gotoh, S. Renals and G. Williams (1999). Named entity tagged language models, *IEEE Intntl Conf. On Acoustics, Speech and Signal Processing*, Phoenix, AZ

X. Huang, A. Acero, H. W. Hon (2001). *Spoken Language Processing*, Prentice Hall, PTR.

D. Janiszek, F. Béchet, R. De Mori (2001). DataAugmentation and language model adapation, *Proc. ICASSP2001*, Salt Lake City, Utah, USA

E.C. Kaiser, M. Johnston and P.A. heeman (1999). Profer: Predictive, robust finite-state parsing for spoken language. *IEEE Intntl Conf. On Acoustics, Speech and Signal Processing*, Phoenix, AZ

T. Kawahara and S. Doshita (1999). Topic independent language model for key-phrase detection and verification, *IEEE Intntl Conf. On Acoustics, Speech and Signal Processing*, Phoenix, AZ

B. Kessler, G. Nunberg and H. Schutze (1997). Automatic detection of text genre, *Proc. of the ACL conference*, Madrid, Spain pp. 32-38.

Klakow (1998). Language-model optimization by mapping of corpora, *IEEE Intl. Conf. on Acoustics, Speech and Signal Processing* Seattle WA

Kuhn and De Mori (1995). The Application of Semantic Classification Trees to Natural Language Understanding, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol PAMI-17, no. 5, Mai 1995, pp. 449-460

K. Mark, Miller M., U. Grenander, S. Abney (1992). Parameter estimation for constrained context-free grammar, *Proc. DARPA SNL*.

Meteer M. and Rohlicek J.R. (1993). Statistical language modeling combining N-gram and context-free grammars. *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, St. Paul, MN, II-37-II-40.

T. Mitchell, R. Keller and S. Kedar-Cabelli (1986). Explanation-based generalization: A unified view. *Machine Learning*, 1, 47:80.

Nasr A., Estève Y., Béchet F., Spriet T., De Mori R. (1999) . *A Language Model Combining N-grams and Stochastic Finite State Automata*, Proc. of Eurospeech'99, Budapest, Volume 5, Page 2175-2178

G. Riccardi (2000). On-line learning of acoustic and lexical units for domain-independent ASR. IEEE Intl. Conf. on Acoustics, Speech and Signal Processing, Istanbul, Turkey

Sadek D., Ferrieux A., Cozannet P., Bretier P., Panaget J., Simonin J. (1996). Effective Human-Computer Cooperative Spoken Dialogue : the AGS Demonstrator. Proc. International Conference on Spoken Language Processing, Philadelphia, USA

Souvignier and A. Kellner (1998). On-line adaptation for language models in spoken dialogue systems, Proc. International Conference on Spoken Language Processing, Sydney, AUS

Y.Y. Wang (2000). A unified context-free grammar and n-gram model for spoken language processing, Proc. IEEE International Conference on Acoustics, Speech and Signal Processing, Istanbul, Turkey