With A Little Help From The Database – Developing Voice-Controlled Directory Information Systems

A. Kellner	**F. Seide**	**B. Rueber**
Philips GmbH	Philips	Philips GmbH
Research Laboratories	Research Laboratories	Research Laboratories
Aachen, Germany	Taipei, Taiwan, ROC	Aachen, Germany

Abstract - Automated directory information is amongst the most challenging applications of automatic speech recognition. In this paper, we present some basic techniques that try to overcome the deficiencies of the speech recognizer by incorporating as much additional knowledge as possible – such as the telephone directory.
We derive a maximum a-posteriori decision rule which explicitly uses the telephone-directory knowledge as well as the dialogue history to improve speech understanding accuracy. The rule allows us to take a combined decision on the joint probability over multiple dialogue turns, which yields good results in combination with spelling.
Our spelling architecture permits continuous spelling of names and uses a context-free grammar to parse common spelling expressions.

We review two different realtime prototypes, on which we evaluated our decision rule. One (PADIS) operates on a small database and one (PADIS-XL) on a database with 130,000 entries.

1 Introduction

For many years, researchers have been investigating the possibilities of using automatic speech recognition in order to fully or partly automate directory assistance services [1, 2, 3, 4]. It is, however, still far beyond the capabilities of today's speech recognition technology to reliably recognize a single spoken name from a list of tens of thousands of candidates.

We have approached this problem by

- explicitly integrating dialogue history and database knowledge into the MAP criterion used for speech understanding;
- the use of spelling. In addition to simple letter-by-letter spelling, the spelling module also accepts descriptive phrases such as "*double T*" or "*M. as in Mike*";
- a combination of system architecture and dialogue strategy that permits to postpone the final decision on the user input until all necessary information is available.

Using these new techniques, we built two realtime prototypes for different scenarios of directory information application:

- **PADIS**[1], a voice controlled automated telephone switchboard for companies of some hundred to a few thousand people. Its mixed-initiative dialogue permits the use of unrestricted natural speech. The caller can request various types of information from the employee directory (around 600 listings), the main service being call transfer [5]. PADIS has been in regular use at our Aachen research lab for over a year.
- **PADIS-XL**, large-scale fully automatic directory information for a city. Our demonstrator handles 130.000 private listings of a medium-size German city [6]. For better performance, we restricted PADIS-XL to a system-driven dialogue with either isolated-word or spelling input.

This paper is organized as follows. After a short overview of the general system architecture, we will review our stochastic framework and present the extended MAP decision rule. Next the spelling module will be discussed. We will then compare both prototypes with respect to the underlying paradigm, the particular realizations of the decision rule, and the search strategies used to achieve realtime operation. Also, some experimental results will be given.

2 System Architecture

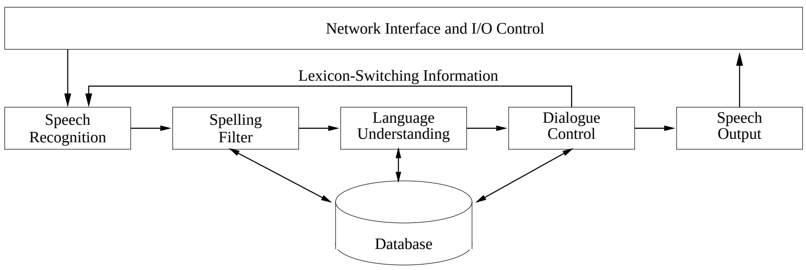

Figure 1: System architecture.

Both prototypes are based on a systems architecture as shown in Fig. 1. The *speech recognizer* delivers a word graph [7]. The *spelling module* scans this graph for letter sequences that form valid names according to a name list. Those name hypotheses are then added to the graph. The *natural-language understanding component* of the system uses an attributed stochastic context-free grammar to parse the word graph for meaningful phrases and derives their meaning. It takes the final decision on the most likely user input. *Dialogue control* follows different strategies for the two systems. It either conducts a mixed-initiative dialogue that permits the user to give information in any order (PADIS), or a system-driven scheme is employed that prompts the user for the desired information items one after the other (PADIS-XL).

[1]Philips Automatic Directory Information System

3 Stochastic Framework, MAP Decision Rule

Let G denote the user's dialogue *goal*. In a directory information application, this goal is to obtain information on a certain directory listing. The user refers to G by specifying a *set of information items* $I = \{I_1, I_2, ..., I_N\}$ like *name*, *first name*, *street* (PADIS-XL), or the desired *service* (PADIS).

We define the speech-understanding task as finding the information-item set $\hat{I}$ that most probably was the one the user formulated when he generated the *acoustic observations* O (maximum-a-posteriori criterion):

$$\hat{I} = \arg\max_I P(I|OS) \tag{1}$$

where S refers to the current *dialogue state*. Typically, a user does not specify all information items in a single turn. In PADIS, we apply the rule turn-wise, i.e. O refers to a single observed utterance, and I denotes the information-item set of this utterance. The dialogue state S contains the system's verified belief, i.e. all items given in previous turns and verified by the system. In PADIS-XL on the other hand, we apply the rule for the whole dialogue, so O refers to the whole sequence $O_1, O_2, ..., O_M$ of user utterances, I is the total information-item set, and the concept of the system state S is not used.

We incorporate W, the underlying word sequence(s) used to formulate I, and the goal G. W and G are unknown, we sum over all possible values:

$$\begin{aligned}\hat{I} &= \arg\max_I \sum_{W,G} p(OWISG) \\ &\approx \arg\max_I \left\{ \max_W p(O|W) \cdot \sum_G P(WIGS) \right\}\end{aligned} \tag{2}$$

The acoustic likelihood $p(O|W)$ is delivered by the recognizer. The second factor, $\sum_G P(WIGS)$ captures the prior knowledge. In its simplest form, it reduces to a language model $P(WI)$, as in [8]. In the systems presented here, $\sum_G P(WIGS)$ also models the prior distribution for the dialogue goal and the dependencies between information items, goal, and dialogue state.

In particular, $\sum_G P(WIGS)$ evaluates to 0 for every I referring to non-existing listings or contradicting the system belief. This way, invalid hypotheses are ruled out directly at decision-rule level. This leads to a major reduction of errors, in particular of those mis-recognitions that make the system appear unintelligent. ("A computer with access to the directory should not come up with a non-existing first/last name combination!")

4 Spelling Filter

Spelling is an important feature in an automatic inquiry system. The recognition accuracy for spelled names is much higher than for spoken names [9]. For a real-life system, we have to handle the common ways people use to spell.

Our spelling module acts as a post-processor to the speech recognizer. It reads a word graph from the recognizer which contains spelled letters and spelling expressions (Fig. 2). As its output, the spelling module creates an extended word graph that contains all spelled words as word hypotheses. This way, spelling becomes transparent for the subsequent language-understanding component. The spelling module operates in a two-stage process:

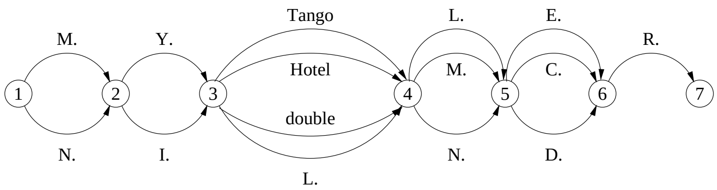

Figure 2: Example word graph for *"M. I. double L. E. R."*.

In the first stage, descriptive phrases in the input are translated into regular letters by parsing the word graph with an attributed stochastic context-free grammar, which contains rules for common spelling alphabets, special characters (*"A. Umlaut"*), and descriptive phrases like "*double T.*" or "*M. as in Mike*". The result of the parse is stored in a pure letter graph (Fig. 3).

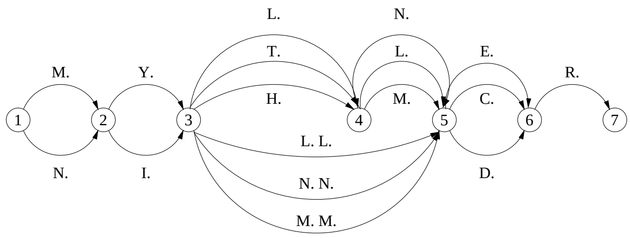

Figure 3: Example letter graph.

In the second stage, the letter graph is searched for letter sequences that form valid words according to a large background word list (e.g. all last names). For every word found, a new word hypothesis is added to the original graph. Its score is computed from the letters' acoustic scores and the language-model score from the stochastic spelling grammar. In our example, the names *Miller*, *Mitler*, and *Milner* would probably be considered valid.

5 The PADIS System

The first prototype system in which the new decision rule was incorporated was PADIS, an automated switchboard system which has been used for internal call handling in our research laboratory since April 1996. PADIS provides telephone and room numbers, e-mail addresses and direct call transfer and

System:	*Hello, this is PADIS. How may I help you?*
Caller:	*I'd like to talk to John.*
System:	*So, you want to talk to John Miller?*
Caller:	*Exactly.*
System:	*The extension 581 is busy right now.*
	Can I do anything else for you?
Caller:	*Oh yes. Can you tell me where Mike is right now?*
System:	*Which Mike is it you want the current location of?*
Caller:	*Uh, he is in the speech recognition group.*
System:	*Mike Anderson from the speech recognition group is currently in the terminal room G202. Should I put you through?*
Caller:	*Yes, please.*
CALL GETS TRANSFERRED	

Figure 4: Example dialogue of the PADIS system.

operates on a database with around 600 entries. It understands natural-language requests in fluently spoken German and conducts a mixed-initiative dialogue (Fig. 4 shows an example). Confidence measures on semantic level are used to skip verification of reliably understood information items [10].

The system is described in detail in [5], where we also report on some user studies carried out on the field-test data. The dialogue success rate in the field test is around 96%.

5.1 Decision Rule

In PADIS, the particular implementation of the prior-knowlegde term $\sum_G P(WIGS)$ is influenced by the following factors:

A. The prior probability for part of the dialogue goal is not found in the directory database but rather trained into the language model. E.g., PADIS' different types of service are asked for using very different wording ("Please connect me to", "What's the e-mail of"), and it would be very difficult to separate its prior probability from the language model.

B. Due to continuous speech input, multiple information items per turn are possible. We make the model assumption that items found in a hypothesis *may not* contradict each other.

C. The decision rule is applied turn by turn, so a useful system belief may exist after the first turn. We assume a hypothesis may not contradict the verified belief.

These factors are addressed by the following means [11]:

- For A, we split the goal G into two disjunct subgoals, the *database part* G_{DB} (requested listing) and the part modelled by the *language model*, G_{LM}. For G_{DB}, the prior distribution is explicitly available: $P(G_{DB})$ reflects how likely the listing G_{DB} is asked for; its value should be provided by the underlying database. For G_{LM} on the other hand, the prior distribution is implicitly modelled by the language model.

- We introduce a template layer for W and I. W^T and I^T are identical to W and I except that every word directly referring to the database is replaced by a placeholder. Thus, the *template language model* $P(W^T I^T)$ models *how* a request is formulated, but not *which* listing is asked for. However, $P(W^T I^T)$ implicitly models the prior distribution of G_{LM}.
- We define a *matching operator* $\delta_{X,Y}$ as

$$\delta_{X,Y} = \begin{cases} 1 & \text{if } X \text{ does not contradict } Y \\ 0 & \text{otherwise.} \end{cases}$$

  The purpose is to test whether X and Y are contradictory, as needed to implement B and C. Note that $\delta_{X,Y}$ with $X = Y = I$ means to test whether an information-item set contradicts itself, for example if I contains two different last names.

With this, the final decision rule for the PADIS system is [11]:

$$\hat{I} \approx \arg\max_I \left\{ \max_W \underbrace{p(O|W)}_{\text{acoustics}} \cdot \underbrace{P(W^T I^T)}_{\text{grammar}} \cdot \underbrace{\sum_{G_{DB}} \delta_{WI,G_{DB}} \cdot P(G_{DB})}_{\text{database knowledge}} \cdot \underbrace{\delta_{I,S}\delta_{I,I}}_{\substack{\text{consistency}\\ \text{constraints}}} \right\} \quad (3)$$

5.2 Search Strategy

This decision rule is to be applied on a whole-sentence level. For this, we developed an N-best algorithm that permits to obtain the top N candidates one by one sorted by their score using a simplified model [12]. In this simplified model, the database constraints and contradiction tests have been left out. For every hypothesis, a sentence score is computed using the full model. Starting with the first best path (simplified model), this procedure is applied repeatedly until the best path (full model) is found.

5.3 Experimental Results

A quantitative evaluation was carried out on real-life data collected from our PADIS prototype. Table 1 shows word and concept error rates (WER, CER). The CER measures errors at the end of the speech-understanding stage. It summarizes attribute substitutions (the concept's value is wrong, e.g. last name replaced by another), concept substitutions (e.g. a last name substituted by a first name), concept insertions and deletions.

Table 1: Results on the PADIS field-test corpus.

Model	WER	CER	Sub_{Attr}	Sub_{Conc}	Ins	Del
Baseline	28.9%	33.4%	768	441	271	527
+ within-utt. constraints	24.6%	27.8%	487	429	231	541
+ dialogue history	24.4%	26.9%	476	375	240	544
Relative improvement	16%	23%	39%	15%	11%	-3%

The first row shows the baseline, the second the results for incorporating within-utterance constraints (database, contradictions), and the third row also including the dialogue history (full model, eq. 3). The total CER reduction is 23%. As expected, the major gain (39%) is in attribute substitutions.

6 The PADIS-XL System

The second system we built is a demonstrator for automated directory information for the city of Aachen, Germany. Its 131,000 listings include 38,608 distinct last names, 9938 first names and 2049 streets. The main focus of that system was not primarily on usability issues but on how to handle the very large search space with today's speech recognition technologies.

Today's technology cannot yet handle such large vocabularies in realtime. This renders a mixed-initiative dialogue quite difficult. Therefore, for PADIS-XL, we chose for a system-driven approach allowing the recognizer's lexicon to be switched to only those words expected in the current turn. To improve recognition accuracy, the user is prompted separately for each information, and is requested to answer in a single item (spoken or continuously spelled).

6.1 Decision Rule

In contrast to the PADIS system, where the decision for $\hat{I}$ was taken for every dialogue turn, we go one step further in PADIS-XL and jointly apply the decision rule to all turns (and drop the dialogue history S):

$$\sum_G P(WIGS) \quad = \quad \sum_G P(W|I) \cdot P(I|G) \cdot P(G)$$

In the system-driven single-word approach, exactly one information item I_i is collected per turn with $\{I_1, I_2, I_3, I_4\} = \{$ *the last name spelled*, *the last name spoken*, *the first name (spoken)*, *the street (spoken)* $\}$. We obtain:

$$\hat{I} \quad \approx \quad \arg\max_I \left\{ \max_W \underbrace{\prod_{i=1}^{M} p(O_i|W_i)}_{\text{acoustics}} \cdot \underbrace{\prod_{i=1}^{M} P(W_i|I_i)}_{\text{grammar}} \cdot \underbrace{\sum_G \delta_{I,G} \cdot P(G)}_{\text{database knowledge}} \right\} \quad (4)$$

The prior $P(G)$ corresponds to $P(G_{DB})$ in PADIS. The matching operator $\delta_{I,G} = P(I|G)$ is 1 if all I_i match their respective G_i, and 0 otherwise. The term $P(W_i|I_i)$ is either a spelled-letter language model or, for non-spelled utterances, the matching operator δ_{W_i,I_i}.

6.2 Search Strategy

Unlike PADIS, the decision rule is applied to the whole dialogue in PADIS-XL. Thus, the search strategy has to consider the whole dialogue and becomes closely linked to the dialogue strategy.

The dialogue aims at reducing the search space with every turn. In the beginning, all listings of the database are possible candidates, so the search space consists of the full database. In the first turn, the user is asked to spell out the desired last name. Although the search space consists of the full database, the recognizer is limited to spelling, i.e. it only has to recognize letters and the words used in descriptive phrases, which can be done in realtime. The search space is then reduced to names found in the spelling graph.

In the subsequent dialogue turns, the recognizer is dynamically configured to recognize only those words (last names, first names, or streets, respectively) that refer to the candidates that have survived the previous turns. It generates a set of word hypotheses, the path scores of which then are combined with the scores of the corresponding candidates. This forms a new candidate list with a joint probability assigned to each candidate.

Candidates not found in the recognizer's output anymore are deleted from the search space. This occurs due to beam-pruning during word-graph generation. In addition, candidates classified as mis-recognized according to a confidence score are discarded. The procedure terminates when only three or less candidates remain, which are then directly presented to the user.

6.3 Experimental Results

For PADIS-XL, we were interested in the gain obtained from combining the score for the spelled and the spoken instance of a name. Since we had no field-test data for last-name recognition at hand, we evaluated this on first-name recognition instead, for which we could use the German SIETILL database ($N = 2$ information items), cf. [9] for similar results on last names. The vocabulary consisted of 37,961 first names from the city of Hamburg, comparable to the last-name list of Aachen (38K) used in our demonstrator.

As in the demonstrator, we dynamically switched the recognizer's lexicon for each utterance to use only the subset of words found in the corresponding letter graph (on average only 15.3, permitting realtime operation).

Table 2: Results on first-name recognition.

Setup	WER
Best spelled alone	20.8%
Best spoken alone (on switched lexicon)	27.7%
Joint decision	14.3%
Relative improvement	31%

Table 2 shows the results. Taking the spelling result as the baseline, a significant gain of about 30% was achieved by using the joint-decision rule taking both the spelling and the spoken probabilities into account. In about 7% of all utterances (about half of the remaining errors), the name actually spoken was not found in the spelling graph.

7 Conclusion And Outlook

We have presented a stochastic framework which can be applied to various types of directory information systems and which achieves a significant decrease (around 30%) in the error rate of a speech understanding system. This stochastic framework was integrated in an architecture for automatic directory information systems that also includes a flexible spelling component. We have presented two different realtime prototypes, which are based on these techniques and have reported experimental results for the different scenarios.

After all, we believe that automating simple directory-assistance requests, in which the caller knows all required information, will become feasible in the near future.

References

[1] M. Lennig, G. Bielby, and J. Massicotte, "Directory Assistance Automation in Bell Canada: Trial Results", *Speech Communication*, 17(3–4):227–234, Nov. 1995.

[2] C. Kamm, C. Shamieh, and S. Singhal, "Speech Recognition Issues For Directory Assistance Applications", *Speech Communication*, 17(3–4):303–311, Nov. 1995.

[3] B. Kaspar, G. Fries, K. Schuhmacher, and A. Wirth, "FAUST – A Directory-Assistance Demonstrator", *Proc. Eurospeech*, pp. 1161–1164, Madrid, Sept. 1995.

[4] S. Whittaker and D. Attwater, "Advanced Speech Applications – The Integration of Speech Technology into Complex Services", *ESCA workshop on Spoken Dialogue Systems – Theory and Application*, pp. 113–116, Vigsø, June 1995.

[5] A. Kellner, B. Rueber, and F. Seide, "A Voice-Controlled Automatic Telephone Switchboard and Directory Information System", *Proc. IVTTA*, pp. 117–120, Basking Ridge, NJ, Sept. 1996.

[6] F. Seide and A. Kellner, "Towards an Automated Directory Information System", *Proc. Eurospeech*, pp. 1327–1330, Rhodes, Greece, Sept. 1997.

[7] M. Oerder and H. Ney, "Word Graphs: An Efficient Interface between Continuous-Speech Recognition and Language Understanding", *Proc. ICASSP*, vol. II, pp. 119–122, Minneapolis, Apr. 1993.

[8] M. Oerder and H. Aust, "A Realtime Prototype of an Automatic Inquiry System", *Proc. ICSLP*, vol. 2, pp. 703–706, Sept. 1994.

[9] M. Meyer and H. Hild, "Recognition of Spoken and Spelled Proper Names", *Proc. Eurospeech*, pp. 1579–1582, Rhodes, Greece, Sept. 1997.

[10] B. Rueber, "Obtaining Confidence Measures from Sentence Probabilities", *Proc. Eurospeech*, vol. 2, pp. 739–742, Rhodes, Greece, Sept. 1997.

[11] F. Seide, B. Rueber, and A. Kellner, "Improving Speech Understanding by Incorporating Database Constraints and Dialogue History", *Proc. ICSLP*, vol. 2, pp. 1017–1020, Philadelphia, PA, Oct. 1996.

[12] B. Tran, F. Seide, and V. Steinbiss, "A Word Graph based N-best Search in Continuous Speech Recognition", *Proc. ICSLP*, vol. 4, pp. 2127–2130, Philadelphia, PA, Oct. 1996.