Strategies for Name Recognition in Automatic Directory Assistance Systems

H. Schramm, B. Rueber, and A. Kellner
{schramm, rueber, kellner}@pfa.research.philips.com
Philips Research Laboratories
P.O. Box 1980, D-52021 Aachen, Germany

Abstract

The commercial viability of automating large scale directory assistance is shown by presenting new results on the recognition of large numbers of different names. Satisfactory recognition performance is achieved by employing a stochastic combination of N-best lists retrieved from multiple user utterances with the telephone database as an additional knowledge source.
The strategy is used in a prototype of a fully automated directory information system which is designed to cover a whole country: After the city has been selected, the user is asked for first and last name of the desired person and, if necessary, also for the street or a spelling of the last name. Confidence measures are used for an optimal dialogue flow.

We present results of different recognition strategies for databases of various sizes with up to 1.3 million entries (city of Berlin). The experiments show that for cooperative users more than 90% of all simple requests can be automated. Despite the fact that in the field a lot of practical problems like database or lexicon management or acquainting users with the new systems have to be overcome, the authors nevertheless deem the technology to be highly relevant for commercial deployment.

Zusammenfassung

Neue Ergebnisse zur Erkennung vieler verschiedener Namen zeigen die kommerzielle Machbarkeit einer automatisierten Fernsprechauskunft im Großen. Dabei erreicht man eine zufriedenstellende Erkennungsgenauigkeit, indem man die N-best Listen mehrerer Benutzeraäußerungen stochastisch kombiniert, wobei die Telefondatenbank als zusätzliche Wissensquelle verwandt wird.
Diese Strategie wird in einem Prototyp einer vollautomatischen Fernsprechauskunft eingesetzt, die für einen landesweiten Einsatz entworfen wurde: Nach Auswahl der Stadt wird der Benutzer nach dem Vor- und Nachnamen der gewünschten Person gefragt, bei Bedarf dann auch noch nach der Straße oder einer Buchstabierung des Nachnamens. Dabei werden Konfidenzmaße zur Optimierung des Dialogverlaufes benutzt.

Wir präsentieren Ergebnisse verschiedener Erkennungsstrategien auf Datenbasen unterschiedlicher Größen bis zu 1,3 Millionen Einträgen (Berlin). Diese Experimente zeigen, dass man bei kooperativen Benutzern mehr als 90% der einfachen Anfragen automatisieren kann. Obwohl in der Anwendung noch etliche praktische Probleme wie Datenbank- oder Lexikonpflege oder eine geeignete Benutzereinführung in die Systeme zu lösen sind, erachten die Autoren diese Technologie als hochinteressant für einen kommerziellen Einsatz.

Keywords:

directory information; joint recognition; confidence; database constraints; large vocabulary; spelling.

1 Introduction

In recent years, the challenging task of automatic directory assistance has had a lot of attention in the speech recognition community. Several demonstrator systems have been set up and some field trials were performed [13, 6, 12, 9, 3].

Quantitative results on recognition performance for various directory sizes and knowledge sources were published by several groups [9, 12, 5, 10, 7, 8].

This paper investigates the problem of complete automation of directory assistance requests for a whole country and presents systematic results on what is the relative value of using all available knowledge sources.

The paper starts out from the demonstrator system for a fully automated directory information for the city of Aachen with 131,000 database listings [12]. Based on this work, a prototype system was designed which, by its hierarchical structure, can handle a complete country.

A dialogue example from this system is shown in Figure 4. In the course of the dialogue, the system takes a combined decision on the joint probability over multiple dialogue turns, using the directory database itself as additional knowledge source. In this way the search space which consists of all 'active' database entries can be reduced step by step [1].

The rest of the paper is organised as follows: Section 2 presents an overview of the system and its components and describes the dialogue design. A systematic evaluation of the approach follows in section 3. There, error rates are given for using joint (redundant) information with and without dynamic lexicon switching. The use of confidence measures for the early detection of problem cases is described in section 4. Finally, section 5 gives our main conclusions.

2 System Overview

2.1 System Architecture

The prototype system consists of a speech recognizer, a spelling filter, a dialogue manager, and a text-to-speech module as shown in Figure 1. As the results in section 3 indicate, the spelling module could possibly be necessary for huge name databases. The dialogue manager provides the language resource manager with the current system state. From this the active vocabulary for the speech recognizer and the spelling filter is generated.

In the Philips system, the speech recognizer does not deliver a single-best sentence-hypothesis for each user utterance, but creates a word graph which contains many

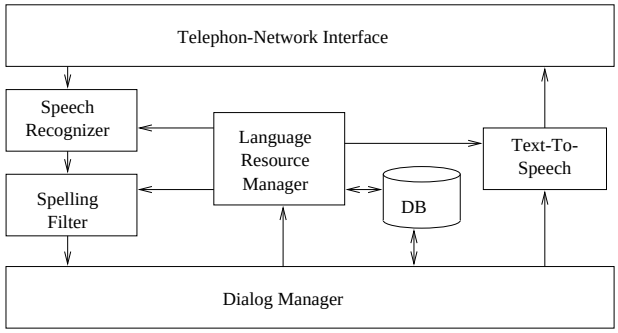

Figure 1: System architecture

different hypotheses and their acoustic scores. This word graph is usually passed to the language understanding module of the system. In our system-driven directory information prototype, the language understanding task is trivial, and therefore, the interpretation of the user input has been integrated into the dialogue manager.

2.2 Speech Recognizer

The speech recognition system used for our experiments was a state-of-the-art continuous density HMM recognizer. This speaker independent telephone-speech decoder works in two different setups for the recognition of spoken respectively spelled words. The switching between these scenarios is done, under control of the dialogue module, by the language resource manager, which also delivers the active vocabulary for the recognition of isolated words.

2.2.1 Isolated Word Recognition

Working in this mode the decoder is restricted to the recognition of a single word per utterance. After a standard MFCC feature extraction, we applied a Linear Discriminant Analysis (LDA) [4] in order to further improve recognition accuracy. The acoustic model consists of 29424 strongly tied context-dependent phonemes which were trained on isolated word telephone speech data. As we focused our interest to the evaluation of the pure acoustical recognition performance, all isolated word experiments have been done without using any language model information. Employing for instance the additional knowlegde of a unigram language model, which was trained on the database, would of course lead to a further error rate improvement.

2.2.2 Spelling Recognizer

The decoder used for the spelling experiments worked with a phoneme set containing two subsets in order to make the

recognition of spelling words like "double" possible. While the first set consisted of phonemes which were trained on continuously spoken timetable inquiry data, the second one comprised 61 context dependent spelling phonemes.

For directing the search in building up the spelling word graph a letter-bigram language model, trained on the telephone directories of the major German cities, was used. This language model information, however, was only used for efficiency reasons and was afterwards removed from the word graphs. Thus, the subsequent processing of the spelling information worked without any language model knowledge.

2.3 Spelling Filter

The recognition accuracy for spelled names is much higher than for spoken names (cf. [10] and section 3). Thus, spelling is an interesting option to obtain additional acoustic input without requiring extra database knowledge from the user.

In a data collection with real users, we saw that in real-life situations people do not always spell a name letter by letter. Instead, they also use expressions like *'double T'* or *'M as in Mike'*. Such descriptive phrases are handled by the spelling module of our system, which acts as post-processor to the speech recognizer.

The spelling module reads a word graph from the recognizer which contains spelled letters and descriptive phrases that are used in 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 modules which can handle spelled words just as if they would have been spoken regularly.

The spelling module operates in a two-stage process:

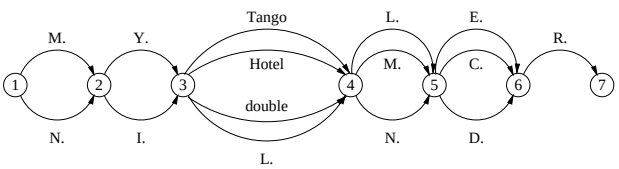

Figure 2: Example word graph for *"M. I. double L. E. R."*.

In the first stage, spelling expressions in the input are identified and 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 approach also permits to handle clarification expressions like "*Meyer with Y.*". The result of the parse is stored in a pure letter graph (Fig. 3). It has the same nodes as the underlying word graph, its arcs are the letters or letter sequences created from the letters and descriptive expressions.

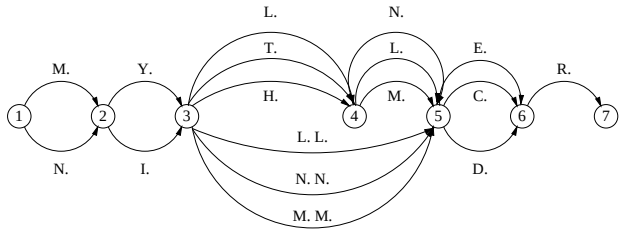

Figure 3: Example letter graph.

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 (depending on the name list).

2.4 Dialogue Strategy

The prototype system follows a hierarchical dialogue strategy (cf. Figure 4): In the first step, the system asks for the city. At this point, only a limited vocabulary containing the largest cities is activated in the recognizer. If the reliability of the best-recognized city falls below a certain threshold, the user is asked to verify the city. If none of the cities was understood with sufficient reliability (using the reliability measure from [11]), the user is asked to spell the city name. At this time, the full city vocabulary is active.

Once the city is selected, the database of this city can be activated. Now, the dialogue aims at reducing the number of active database entries with every turn [1, 12]. In the beginning, the search space consists of all directory listings of the selected city. The system starts by asking for the desired last name. The search space is then reduced to only those database entries for which the name was found in the word-graph[1]. In the subsequent dialogue turns, the recognizer is dynamically configured to recognize only those words (first names, or streets, respectively) that refer to active database entries.

In each turn, the scores of the recognized hypotheses are combined with the scores obtained so far for the corresponding database candidates. This forms a candidate list with a joint probability assigned to each candidate.

[1] Note that a graph is just an N-best list of word hypotheses in case of an isolated word recognition.

System:	Hi, this is the automated directory information.
	From which city do you want to have a listing?
User:	**Aachen.**
System:	Please say the last name.
User:	**Feuerstein.**
System:	Please say the first name.
User:	**Fred.**
System:	Do you mean Fred Feuerstein, Rosenweg?
User:	**Yes.**
System:	The telephone number is ....
	Should I put you through?
	...

Figure 4: Dialogue Example

Confidence measures are then employed to decide on the further flow of dialogue. In that, the idea is to only ask for further information (street or spelling) as long as the joined recognition is not precise enough. As soon as there are only three or less reliable candidates left in the search space, these are presented to the user.

In addition to shaping the flow of dialogue, confidence measures also indicate problem situations requiring error recovery. Natural choices are requesting turn repetitions or additional spellings (of e.g. the first name). Of course, as a final fall-back, the call can be routed to a human operator.

Another option for the design of the dialogue (especially for very large databases), is to ask the user first to spell the last name. As spelling recognizers are much more accurate (cf. section 3) the size of the employed candidate lists can be largely reduced, resulting in a much better computing efficiency. Of course, this implies the user to give a correct spelling of the full last name. As a benefit, then, in most cases the street is not needed any more.

To allow for optimum recognizer performance, the dialogue was deliberately designed in a quite stringent, completely system-driven fashion. But, of course, the user needs a minimum amount of initiative to e.g. express that he does not know a specific item, needs help how to proceed, wishes to restart the dialogue, or simply wants to be transferred to a human operator. Therefore, the system is designed to understand the appropriate commands at every point in the dialogue.

Technically, the complete dialogue behaviour can be configured with a simple C-like dialogue description language which is based on Philips' HDDL [2]. This has considerable advantages to the hard-coded alternative if it comes to system changes or new applications.

3 Results

In this section, the recognizer's ability for the task of large scale directory assistance will be assessed. For that, after explaining the general setup, we present results achieved in our latest combined recognition experiments. Please note that, as already mentioned in 2.2, all results presented were achieved without using any language model information.

3.1 Recognition Experiment Setup

A telephone database of directory assistance inquiries comprising 676 different speakers all over Germany has been collected in the following manner: By various advertisements people were asked to call up a data collection system which prompted them for speaking and spelling their last, first, street, and city names.

This data was used as test set in our experiments. Artificial telephone directories of varying sizes were created using the telephone directory of Berlin, Germany's biggest city with about 1.3 million database entries, in the following way:

- All directories include the test data.
- Then, different percentages of Berlin were added to them as a background list by selecting every n-th entry of the original Berlin directory. So, e.g. "10% Berlin" consists of the test data plus every 10-th entry of Berlin.

3.2 Joint Recognition

As a participant in a directory assistance database is in general characterized by two or more information items (normally the last, first, and street name if the city is already known) the task calls for the combination of more than one knowledge source.

Within the non-spelling experiments presented in this section, we focused our interest to the recognition of at least two information items as from the database's point of view, the majority of all requests is still ambiguous after a single last name turn. Moreover, the acoustical information of an additional, for instance first name turn, helps to further improve recognition accuracy without overly lengthening the dialog. Otherwise, in order to especially point out the effect of an additional spelling step, results for the corresponding experiments are presented after each dialogue turn.

The following alternative joint recognition scenarios were studied [12]:

- SEP: separately recognizing each name category for generation of N-best lists which are only afterwards combined.
- HIER: hierarchical recognition, i.e. starting out with the recognition result of one name category, successively restricting the active lexicon for all subsequent recognition steps as to include only the candidates left over so far.

In both scenarios, combined N-best lists were computed by a standard weighted score addition: Let $sc_i^{(1)}$ be the score of an item i in N-best list 1 and $sc_j^{(2)}$ the score of its matching entry j in N-best list 2, i.e. the one where the combination of the two refers to a valid database entry. Then the score $sc_{i,j}^{(1,2)}$ of the combined entry in the combined N-best list is computed by

$$sc_{i,j}^{(1,2)} = sc_i^{(1)} + \alpha \cdot sc_j^{(2)} \quad (1)$$

The weighting factor α has been optimized on a cross-validation corpus and $\alpha = 1$ turned out to be a reasonable choice.

For the recognition setup, we chose the scenario closest to the human operator service, i.e. assuming that the city already has been determined, we start out with the last name. Then, subsequent questions are posed for first and street name. Even if the scenario of starting with the last name is not optimal from the recognition performance's point of view (which would call for starting with the street name), it may be of a greater practical performance as many users will not know the street name of the person they are asking for. In an alternative scenario a complete spelling of the last name is employed as the entrance step.

Now, Tables 1 – 4 show the first-best, 3-best and graph error rates as well as the amount of safe rejections after each HIER respectively SEP combination step for the databases "100% Berlin" and "10% Berlin". Here, the percentage of safe rejections is the number of cases in which the intersection of all combined N-best lists is empty, i.e. in these cases the system knows that it did not understand the user.

The left column of the tables indicates the combination turn. It has to be noted that with increasing level of combination information the number of recognition units increases. Thus, whereas e.g. for "100% Berlin" the recognition inventory for the spelled and spoken last name recognition consist of 'only' the 189,352 different last names of Berlin, we have to deal with 1,263,957 different recognition units in case of a combined recognition of last, first, and street name, that is all occurring combinations.

Of course, the increase in lexicon size counteracts the growing amount of acoustical knowledge gained by the combinations. To keep the graph error rate low, which finally determines the first-best errors, this requires a careful trade-off in choosing the pruning thresholds versus the computing power spent. We e.g. observe for the HIER scenario "100% Berlin" without initial spelling step an increase in GER from 0.9% to 1.2% at a final first-best error rate of 1.9% (see Table 1).

From the figures in Tables 1 – 4, the following observations can be drawn:

- By avoiding some pruning errors the hierarchical recognition HIER outperforms the SEP scenario only slightly. For e.g. "10% Berlin" without initial spelling step we observe a 1.4% absolutely better error rate for the HIER scenario after the street name turn. But the hierarchical approach is computationally much more efficient as for all but the first recognition step only small lexicons (as compared to the SEP recognitions) have to be employed. On the other hand, the SEP scenario is an interesting architecture alternative minimizing the database accesses and exploiting all available acoustical knowledge for the combination of the information items [1]. The latter is especially important for the computation of confidence measures (see item about safe rejections below).
- Relating word and graph error rates to the database size and the amount of combination information it is obvious that with every database the recognition is able to achieve very low error rates as soon as enough knowledge sources are available. I.e. at this stage the remaining errors are completely determined by graph errors. Thus, an ER below 10% is achieved for "10% Berlin" with spoken last plus first name while for "100% Berlin" one additionally needs e.g. the street or a spelling.
- Starting with the spoken last and first name, the first-best error rate for "100% Berlin" is about 16%. To further increase the accuracy, an additional last name spelling step could be employed leading to a first-best error rate of 4%. But as spelling only supplies a different acoustical representation of an already known item (the last name), the result can still be ambiguous. In case of different participants with the same first and last name, a further turn is necessary, for instance a street name turn. A probably better scenario would be to start asking for the street name information directly after the first and last name turn. If the street name is known by the caller, the resulting error rate is with 3.3% slightly better than with spelling. Furthermore the resulting database entries are most probably no more ambiguous. If the caller does not know the street name, a spelling turn could still be used as a fall-back. But in this case, the system as well as a human operator has no chance to eliminate the remaining ambiguity.

Table 1: First best, 3-best, graph error rates, and safe rejections of combinations for 100% of Berlin (1,280,342 db entries) without spelling

turn	# rec. units	comb. ER SEP				comb. ER HIER			
		ER	3-best ER	GER	Rej	ER	3-best ER	GER	Rej
last name + first name	961,894	15.8%	9.6%	1.6%	0.3%	15.8%	9.6%	0.9%	0%
+ street name	1,263,957	3.3%	2.8%	2.8%	2.2%	1.9%	1.5%	1.2%	0%

Table 2: First best, 3-best, graph error rates, and safe rejections of combinations for 100% of Berlin (1,280,342 db entries) with spelling

turn	# rec. units	comb. ER SEP				comb. ER HIER			
		ER	3-best ER	GER	Rej	ER	3-best ER	GER	Rej
spelled last name	189,352	14.4%	4.3%	1.0%	0.2%	14.4%	4.3%	1.0%	0.2%
+ last name	189,352	7.5%	2.7%	1.9%	0.4%	7.3%	2.4%	1.2%	0.3%
+ first name	961,894	4.3%	3.4%	2.7%	1.5%	3.7%	2.4%	1.2%	0.3%
+ street name	1,263,957	4.0%	3.9%	3.9%	3.7%	1.8%	1.6%	1.5%	0.3%

Table 3: First best, 3-best, graph error rates, and safe rejections of combinations for 10% of Berlin (128,642 db entries) without spelling

turn	# rec. units	comb. ER SEP				comb. ER HIER			
		ER	3-best ER	GER	Rej	ER	3-best ER	GER	Rej
last name + first name	123,567	8.3%	3.7%	1.2%	0.4%	8.3%	3.7%	0.6%	0.0%
+ street name	128,608	2.7%	2.4%	2.4%	1.9%	1.3%	1.0%	0.9%	0.0%

Table 4: First best, 3-best, graph error rates, and safe rejections of combinations for 10% of Berlin (128,642 db entries) with spelling

turn	# rec. units	comb. ER SEP				comb. ER HIER			
		ER	3-best ER	GER	Rej	ER	3-best ER	GER	Rej
spelled last name	56,993	7.8%	3.0%	1.2%	0.2%	7.8%	3.0%	1.2%	0.2%
+ last name	56,993	5.6%	2.1%	1.8%	0.6%	5.6%	2.1%	1.5%	0.2%
+ first name	123,567	2.8%	2.7%	2.4%	1.5%	2.4%	1.9%	1.5%	0.2%
+ street name	128,608	3.7%	3.6%	3.6%	3.3%	2.1%	1.9%	1.8%	0.2%

- Even for "100% Berlin", in the SEP recognition, a substantial part of the 3.3% remaining errors can be safely rejected as the resulting combined n-best lists are empty. Thus, there are less than 1% real errors which need to be treated in a further dialogue step.
- The 3-best error rate, probably the most relevant error rate for this application, is below 10% for "100% Berlin", even without any spelling or street name turn. For "10% Berlin", corresponding to a medium size city, we achieve an accuracy of even more than 95%.

Generally it can be said that, for bigger tasks, the problem of generating word graphs of high enough quality in each recognition step is substantial. Moreover, each combination step should, if possible, be chosen in a way that the increased amount of recognition units is in accordance with the additional acoustical knowledge. I.e. the gain in acoustical knowledge must outbalance the loss in recognition security which is due to the bigger recognition inventory. Finally, as already stated before, it would of course be possible to further improve the results by using language model information.

4 Confidence Measures for Early Detection of Misrecognitions

Besides the empty N-best lists of the SEP recognition explicit confidence measures can be used to judge the accuracy of the progressing dialogue. Thus, in problem situations appropriate error recovery strategies can be initiated.

To optimize speech recognition performance we deliberately chose the system architecture to prompt the user in separate turns for each name component. This, of course, introduces the disadvantage that the user has to go through several dialogue turns, a fact which becomes even more annoying in the case of a dialogue failure. The numbers in section 3 show, that these failures are not negligible if the user only knows first and last name of the person in a big city.

To address this undesired situation of failures after lengthy dialogues the question naturally arises if confidence measures for the transaction success can already be computed at an early stage of the call. As a first attempt in this direction we here give some preliminary results on computing confidence measures already after the last-name turn for the correctness of the later on combination of last and first name. I.e., we investigate the following scenario:

- The dialogue aim is to recognize the full name (first + last name) correctly (and then output the corresponding phone number).
- For that, the system first prompts for the last name, then, in a second turn, for the first name.
- After that second turn, first and last name are combined (using the separate-combination (SEP) strategy of section 3.2).
- The dialogue is considered successful only if this first and last-name combination is correct.
- The confidence measure should be computed already after the first turn (i.e. after the last name) to e.g. allow an early operator fall back to avoid customer frustration.

We present Receiver Operating Characteristics (ROC curves) for the "100% of Berlin" scenario. An ROC curve plots, at various values of the confidence rejection threshold, the number of (not rejected) accurately recognized items versus the number of (not rejected) falsely recognized ones. As such, it is a standard criterion for assessing the quality of a confidence measure.

The main confidence measure investigated is the standard a posteriori probability of the recognized name in the N-best list of its competitors [11]:

$$p_1 = \frac{e^{-\lambda \cdot sc_1}}{\sum_{i=1}^{N} e^{-\lambda \cdot sc_i}}, \tag{2}$$

where N is the length of the N-best list, sc_i the utterance score of the ith-hypothesis in the N-best list and λ is an empirical scaling factor chosen to $\lambda = 1$ in this investigation.

Figure 5 presents two ROC curves, both for the correctness of the full name (separate first- plus last-name combination):

1. the lower one computed from the last-name turn only,
2. the upper one, for comparison reasons, computed from the combined last and first-name turns, i.e. from the N-best list of the full names.

As can be seen from the lower curve in this figure the confidence computed from the last-name turn only already carries a considerable portion of information on the correctness of the full name. Interesting operating points are

- 11% false alarms at 79% accuracy (a 31% relative false-alarm reduction at an only 6% relative accuracy reduction),
- or 4% false alarms at 55% accuracy (a 75% relative false-alarm reduction at a 35% relative accuracy reduction).

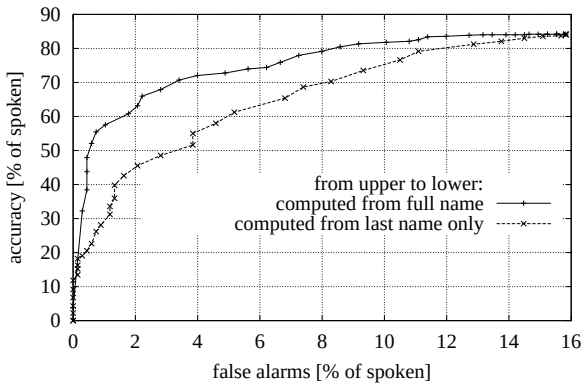

Figure 5: ROC curves for correctness of full name.

But, not surprisingly, there is also a considerable loss in confidence information in not knowing the first name turn: The confidence computed from the full name, i.e. the upper curve of the figure lies, especially in the low false-alarm rate region, significantly above the one from the last-name turn only, i.e. the lower curve. Thus, the confidence from the full name still offers a satisfactory operating point at very low false-alarm rates, e.g.

- 1% false alarms at 58% accuracy (a 94% relative false-alarm reduction at a 31% relative accuracy reduction),

a point where the last-name confidence is nearly a factor of 2 worse in accuracy (30% as compared to 58%).

5 Conclusions

In this paper we presented the results achieved in our latest directory assistance experiments. From these results we can say that, assuming cooperative users and simple requests, the task of directory assistance can be fully automated at a high accuracy even for very large databases.

For medium size cities (130k database entries) automation with a first-best accuracy of about 92% can be achieved by only using the information of the spoken last and first name. There is, from the recognition performance's point of view, especially no need for an additional spelling or street name turn. If it is moreover acceptable to present the 3-best result to the caller, the accuracy increases to 96%. In this case, also big cities like Berlin with 1.3 million participants can be fully automated with an accuracy of more than 90% only with the information of the spoken first and last name. If also the street name is known by the caller, the first-best recognition accuracy is about 97%. In case the caller does not know the street name, spelling is an interesting possibility to further increase accuracy. With an additional spelling turn we achieved an accuracy of 95% on the city of Berlin.

Confidence measures have been shown to allow a good prediction on the accuracy of the progressing dialogue. Especially, they give the system developer the design option to trade-off between the amount of calls classified as problematic and the rate of remaining undetected errors. As an example, for a high customer quality system for the "100% Berlin" case, knowing only first and last name, one may choose to hand-off 42% of the calls to human operators while the remaining 58% can be serviced nearly without any remaining failure (1% absolute error only).

To allow for further flexibility in system design, we have demonstrated that such confidence measures might already be computed at a very early stage in the call. This allows to design for even higher customer satisfaction.

Taking a technology from the laboratory into the field, of course, introduces a lot of practical problems. Today's human-human dialogues show a whole bunch of spontaneous speech effects, unknown information items, wrong pronunciations, incomplete databases, to mention just a few. Nevertheless, these problems appear to be solvable, by improving databases and lexicons, employ confidence measures, and, maybe the most important, by carefully introducing the new systems and their benefits to the users.

Therefore, in the opinion of the authors, a degree of technological progress has been achieved which clearly shows the commercial viability of automating directory assistance services on the large scale.

References

[1] David J. Attwater and Steve J. Whittaker. Issues in large-vocabulary interactive speech systems. *BT Technol J*, 14(1):177–186, January 1996.

[2] H. Aust and M. Oerder. Dialogue control in automatic inquiry systems. In *ESCA Workshop on Spoken Dialogue Systems*, pages 121–124, Vigsø, Denmark, Jun. 1995.

[3] R. Billi, F. Canavesio, and C. Rullent. Automation of Telecom Italia directory assistance service: Field trial results. In *IVTTA*, pages 11–16, Torino, Italy, September 1998.

[4] R. Haeb-Umbach and H. Ney. Linear discriminant analysis for improved large vocabulary continuous speech recognition. In *Proc. ICASSP*, volume I, pages 13–16, San Francisco, CA, Mar. 1992.

[5] C.A. Kamm, C.R. Shamieh, and S. Singhal. Speech recognition issues for directory assistance applications. *Speech Communication*, 17(3–4):303–311, November 1995.

[6] B. Kaspar, G. Fries, K. Schuhmacher, and A. Wirth. Faust – a directory-assistance demonstrator. In *Proc. EUROSPEECH*, pages 1161–1164, Madrid, September 1995.

[7] A. Kellner, B. Rueber, and H. Schramm. Strategies for name recognition in automatic directory assistance systems. In *Proc. IVTTA*, pages 21–26, Torino, Italy, September 1998.

[8] A. Kellner, B. Rueber, and H. Schramm. Using combined decisions and confidence measures for name recognition in automatic directory assistance systems. In *Proc. ICSLP*, volume 7, pages 2859–2862, Sydney, Australia, Dec. 1998.

[9] Matthew Lennig and Gregory Bielby. Directory assistance automation in Bell Canada: Trial results. In *Proc. IVTTA*, pages 9–13, Kyoto, Japan, Sep. 26–27 1994.

[10] Michael Meyer and Hermann Hild. Recognition of spoken and spelled proper names. In *Proc. EUROSPEECH*, volume 3, pages 1579–1582, Rhodes, Greece, September 1997.

[11] Bernhard Rueber. Obtaining confidence measures from sentence probabilities. In *Proc. EUROSPEECH*, volume 2, pages 739–742, Rhodes, Greece, September 1997.

[12] Frank Seide and Andreas Kellner. Towards an automated directory information system. In *Proc. EUROSPEECH*, volume 3, pages 1327–1330, Rhodes, Greece, September 1997.

[13] S.J. Whittaker and D.J. Attwater. Advanced speech applications – the integration of speech technology into complex services. In *ESCA workshop on Spoken Dialogue Systems – Theory and Application*, pages 113–116, Vigsø, June 1995.