USING COMBINED DECISIONS AND CONFIDENCE MEASURES FOR NAME RECOGNITION IN AUTOMATIC DIRECTORY ASSISTANCE SYSTEMS

A. Kellner, B. Rueber, and H. Schramm
{kellner, rueber, schramm}@pfa.research.philips.com
Philips Research Laboratories
Weisshausstrasse 2, D-52066 Aachen, Germany

ABSTRACT

Directory assistance systems are amongst the most challenging applications of speech recognition. Today, complete automation of the service fails because of the lacking accuracy of current speech recognizers, which are simply not able to differentiate between hundreds of thousands or even millions of different names occurring in large cities. In this paper, we show that this situation can be remedied by systematically combining all available knowledge sources (last names, first names, street names, partly including their spelled versions) in a statistically optimal way. Especially designed confidence measures for N-best lists are proposed to detect misrecognized turns.

Applying these techniques in a hierarchical setup is judged as the enabling step for automating large scale directory assistance. In first experiments, we e.g. are able to service 72% of the inquiries for a database of 1.3 million entries with a remaining error rate of only 6% (or 62% with an error rate of 2%).

1. Introduction

Fully automatic directory assistance is still one of the big challenges in the field of spoken dialog systems. A lot of research has been carried out in this area in the recent years (cf. [1, 2, 3] a.o.), but until today, there is still no system in the market.

The main technological problem in this application is the recognition of names from a list of several 100.000 or more candidates [3, 4, 5] .

We have developed a prototype which by its hierarchical structure is capable of handling a complete country. In this system, the user first has to select a city (from a list of 10.000 German towns) and is then asked to give a number of information items about the desired person. Starting from a spelled last name, the system asks for the spoken last name, first name, and street respectively and takes a combined decision on the joint probability over all dialog turns [6]. The telephone database itself is used as additional knowledge source [5].

After a short system overview, results of different combination strategies are presented in section 3. In section 4., we show how many of the system's misrecognitions can be detected automatically by using confidence measures.

2. System Overview

2.1. System Architecture

The prototype system [6] consists of a speech recognizer, a spelling filter, a dialog manager, and a text-to-speech module.

Depending on the current dialog state and the set of active database entries, the vocabulary for the speech recognizer and the background wordlist of the spelling module can be restricted to only those words which are expected in the current situation. In addition to those words, a set of command words is always active which allow the user to take initiative in the dialog like asking for help, repetition, or a restart of the dialog.

A detailed description of the dialog control and the knowledge update in the system can be found in [6] and [5] respectively.

2.2. Speech Recognizer

The experiments presented in this paper were obtained using a speaker independent telephone-speech decoder. This state-of-the-art continuous density HMM recognizer works with two different setups for the recognition of spoken respectively spelled words. The switching between these setups is done by the language resource manager which also provides the list of active lexicon entries for every dialog state.

Single Word Recognition: In this mode, the decoder is restricted to the recognition of a single word per utterance. An inventory of 3502 strongly tied context-dependent phonemes was used. Due to a lack of appropriate (i.e. isolated word) speech data, we trained this phoneme set on a large German spontaneous speech database, consisting of 33081 utterances (12.1h non-silence) of train-schedule inquiries. This mismatch causes, of course, an increase in word error rate which, however, does not influence the qualitative results presented here.

Spelling Recognizer: In order to allow for the recognition of spelling words like 'double', our spelling recognizer worked with a phoneme set composed of two subsets. The first one consisted of the phonemes used for the isolated word recognition and the second comprised 61 context dependent spelling phonemes. The latter were trained on 1637 spelled first names, words, and random letter sequences (1.2h non-silence frames), taken from the German telephone database SPEECHDAT.

2.3. Spelling Filter

Spelling is an essential feature for handling large vocabularies. Our studies have shown that in real-life situations, people tend to use descriptive phrases like 'double T' or 'M as in Mike' rather than simply spelling a name letter by letter.

Our system therefore uses a spelling filter which acts as a postprocessor to the speech recognizer. This filter first detects spelling expressions using a context-free grammar and transforms them into generic letter sequences. Then, a background wordlist, provided by the language resource manager, is used to identify valid names in the letter graph. The spelling filter is described in more detail in [5].

3. Recognition Experiments

The telephone data we used as test set in our experiments consisted of directory assistance inquiries spoken by 676 different speakers from all over Germany. The test-set database entries were merged together with entries taken from the Berlin directory to compose an artificial telephone directory. This directory comprises 56,993 last names, 123,567 last-name/first-name combinations, and 128,608 last-name/first-name/street-name combinations.

It turned out, that the word error rates are much too high to appear useful for a practical application. Therefore, the following alternative joint recognition scenarios were studied in order to improve the name recognition performance [6]:

1. SEP: separately recognizing each name category for generation of N-best lists which are only afterwards combined,
2. SEP*: same as 1, but (as a control experiment to assess the importance of pruning errors) always artificially adding the spoken word to the word graphs (by using forced alignment),
3. 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 all these scenarios, combined N-best lists were computed by a standard weighted score addition: Let sc_1 be the score of an item in N-best list 1 and sc_2 the score of its matching entry in N-best list 2, i.e. the one where the combination of the two refers to a valid database entry. Then the score $sc_{1,2}$ of the combined entry in the combined N-best list is computed by

$$sc_{1,2} = sc_1 + \alpha * sc_2$$

The weighting factor α has been optimized on a cross-validation corpus and $\alpha = 1.0$ turned out to be a reasonable choice.

As a scenario for the recognition setup, we chose the same as in our online demonstrator, where, assuming that the city already has been determined, the dialog starts out with the last name. Then, subsequent questions are posed for first and street name. Furthermore, to avoid problems caused by hardly distinguishable last names a complete spelling of the last name is employed as the entrance step.

At this point, we only give an overview of the recognition experiments carried out. For a more detailed discussion together with a complete presentation of the results we obtained, see [6].

Table 1 shows, besides the baseline result of spelling only, the word and graph error rates and the percentage of safe rejections for the different combination methods. Here, the percentage of safe rejections is the number of cases in which the intersection of all combined N-best lists is empty. In these cases the system 'knows' that it did not understand the user correctly.

Level of Combination	Method	WER[%]	GER[%]	Rej[%]
last name spelled alone	SEP	20.4	15.2	11.0
previous + last name spoken	SEP	48.4	46.5	41.8
	SEP*	21.3	18.5	15.5
	HIER	19.5	15.2	11.0
previous + first name spoken	SEP	52.5	50.4	45.0
	SEP*	23.2	20.9	17.6
	HIER	18.5	15.5	11.0
previous + street spoken	SEP	55.5	55.2	53.0
	SEP*	24.3	24.0	22.6
	HIER	16.9	16.7	11.7

Table 1: Error and rejection rates for the different combination methods.

From this table, it can be seen that the hierarchical recognition is a powerful method to avoid the search problems observed for a recognition on a full, i.e. non-restricted, lexicon. This approach already leads, together with the safe rejections, to a very low level of false information. The following section gives first results on how the remaining misrecognitions can be detected by employing especially designed confidence measures.

4. Confidence Measures for N-best Lists

In a setup where the final interpretation of the user's answers is only obtained after combining the N-best lists of all his utterances a completely new situation for a confidence tagger arises: For the usability of the recognized N-best list of a particular utterance not the correctness of its first best candidate matters but its contribution to the final combination. Thus, what is looked for are confidence measures correlating with the probability that the N-best list a) at all contains the right candidate and b) contains it with a recognition score which is beneficial for the combination with the other lists.

In the following, a variety of possibly useful confidence measures are proposed and first results on their performance are presented. For that, the most critical step of the hierarchical recognition scenario (HIER) of the preceding section is chosen: the recognition of the spelled last name. As background lexicon, the complete directory of Berlin is employed, comprising 1,263,957 different first-/last-/street-name combinations.

In the HIER recognition setup, the spelling step dramatically reduces the size of the active vocabulary for all subsequent recognitions. Consequently, they can be expected to be rather simple (and thus reliable) on their own. Now, if any problems occur during these recognitions they are probably due to a bad original spelling list. Therefore, also these turns may be employed for computing the spelling list's confidence (subsection 4.4.). Of course, it is desirable to detect bad spelling turns as soon as possible, i.e. ideally using the spelling only (subsection 4.3.).

4.1. Evaluation Methods for Confidence Measures of N-best Lists

As said above the quality of a confidence measure for an N-best list has to be evaluated with respect to its usefulness for recognizing the correct item in the final combination of all turns. But in the final combination all N-best lists take part which considerably complicates the evaluation of the confidence of the spelling list alone. Therefore, one may resort to the already mentioned assumption that all subsequent recognitions (on the small lexica) are reliable, i.e. all problems in finding the correct item after combination are solely attributed to bad spelling lists.

Figure 1 now displays the ROC-curves for two of the proposed confidence measures that will be explained below (subsection 4.2.). For each confidence measure, 3 curves are shown: a) for the left most, "correctness" of an N-best list is defined as the N-best list containing the correct spelling, b) in the middle curve, an N-best list is "correct" if the final combination recognizes the correct last name as its (first) best, and c) in the right most curve, "correct" is if the final combination recognizes the complete correct item, i.e. the correct first-/last-/street-name-combination as its (first) best.

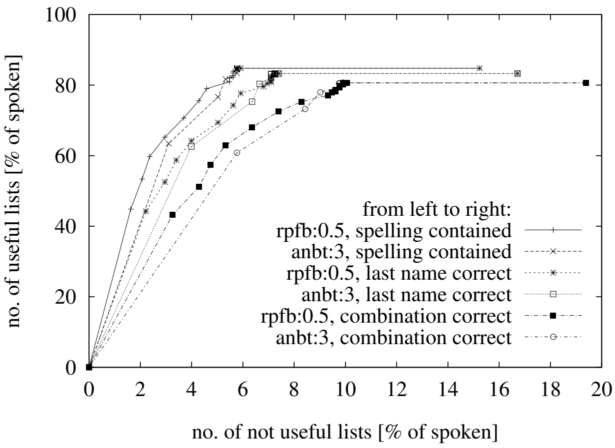

Figure 1: Evaluating confidence measures for N-best lists (see text).

ROC is a shorthand for "Receiver Operating Characteristic" and is a curve plotting the recognizer's accuracy (the number of N-best lists useful for the final combination) versus its false-alarm rate (number of not useful lists). Such curves are used a.o. as a standard criterium for assessing the quality of a confidence measure (see below and cf. e.g. [7]).

What can be seen from Figure 1 is that the qualitative behavior of the ROC-curves of different confidence measures, e.g. their relative position, is the same in all 3 evaluation scenarios shown. So, please notice that already the graph errors of the original spelling N-best list, i.e. the fact if it contains the correct candidate, conveys the main information on its usability for the subsequent combination. Of course, this a) reflects the generation process of these lists which are obtained by likelihood pruning during the search process (and not by any arbitrary cutting), and b) supports the above assumption that the quality of the spelling lists is the main factor determining the success of the final combination. So, because it is the genuine aim of the recognition, we present all following ROC-curves for the final-combination evaluation scenario, i.e. the right most curves (scenario c) in Figure 1.

4.2. Proposed Confidence Measures

The following set of confidence measures is partly theoretically motivated as a generalization of the a posteriori probability of a recognized sentence. As such their basic idea is related to the log-likelihood ratio scoring which was first proposed in [7] and further elaborated in [8].

All measures employ the concept of the set of first bests which are the first best candidates in the N-best list. Then, some feature of this set of first bests is taken as confidence measure. In detail, the definitions are as follows:

rpfb:n$<num>$ This is the a posteriori probability of the set of the $<num>$-best candidates of the list computed by renormalizing the total probability of the list to 1.

rpfb:$<\Delta s>$ Similar to the above but as set of first bests all candidates are taken whose (recognition) score is at most $<\Delta s>$ above that of the (first) best candidate.

rnbt:$<\Delta s>$ The set of first bests is defined as for "rpfb: $<\Delta s>$" but instead of its a posteriori probability, the number of candidates in this set in relation to the total number of candidates in the list (i.e. the corresponding quotient) is considered.

anbt:$<\Delta s>$ As in "rnbt:$<\Delta s>$" but instead of the relative number, the absolute number of candidates in the set of first bests is taken.

Clearly, all these measures relate to the intuitive idea that candidates which are likely to survive in the final combination do not have a large score difference to the best candidate. Furthermore, in order to have enough discriminative properties, useful N-best lists should not have too many of such candidates.

4.3. Performance Comparison

Of course, the ideal function of a confidence measure is to reject all not useful lists while keeping all useful ones. Thus, a measure is the better the more its ROC-curve bends to the left.

Now, for the spelling N-best lists, due to the a priori constraint that only known last names were spelled we observe a considerable number of empty spelling lists. Of course, at first these empty graphs are tagged as unreliable, lowering the number of not useful lists $n_{nu} = 19.4\%$ to $n_{nu} = 10.1\%$ while keeping the number of useful ones at $n_u = 80.6\%$.

Therefore, Figure 2 only shows the interesting part of the ROC-curves of the proposed measures for $n_{nu} \leq 10.1\%$. These curves prove the benefit of the measures by comparing their leftward bend to the diagonal line behavior of a random tagger also depicted in Figure 2.

Of course, the parameter values of the confidence measures in Figure 2 have been chosen at their optimal values. But these optima are quite broad and easy to find by collecting simple score statistics of the N-best lists. Furthermore, they are robust while transferring them to new corpora of the same application.

Looking at Figure 2, rpfb:$<\Delta s>$ and rnbt:$<\Delta s>$ are the best taggers. Intuitively (and theoretically), this appears quite satisfac-

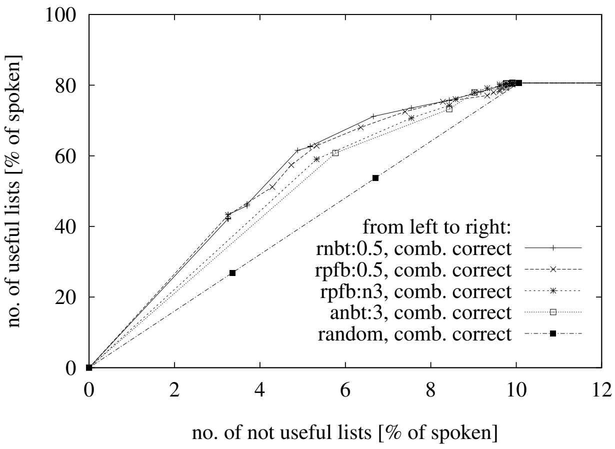

Figure 2: ROC-curves of the proposed confidence measures.

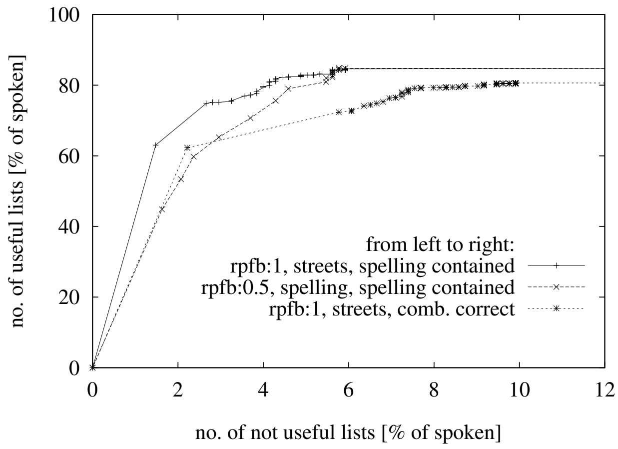

Figure 3: ROC-curves of the final street name recognition. (middle curve reproduced from Figure 1 for comparison only)

tory as the absolute score distance of a candidate to the (first) best obviously has a strong influence on this candidate's ability to be recognized by the subsequent combination steps. Furthermore, the a posteriori probability of the set of first bests, which is in practice closely related with the relative number of its members, measures the trust the spelling recognition already puts in them and thus should indicate if they are actually correct.

The advantage of the rnbt:$<\Delta s>$ over the random tagger may be expressed in a single number by noting that the area under its ROC-curve (in the interesting area $0 \leq n_{nu} \leq 10.1\%$) is 30% larger than that of the random tagger. A useful operating point for the confidence threshold seems to be at $n_{nu} = 4.9\%$ where $n_u = 61.5\%$ and thus 57% (relatively) larger than that of the random tagger.

4.4. Using the Subsequent Turns

As already explained at the beginning of this section the spelling lists, even after being accepted at first, might still be rejected if problems show up in the further turns. These rejections become very reliable. For illustration, Figure 3 applies the "rpfb:1" measure to the final street name turn (without any beforehand rejections) and shows how securely the correctness of the final combination of all turns can be accessed (right curve). Actually, as claimed, most of the false alarms are due to graph errors of the spelling lists. This can be seen from the left curve in Figure 3, where for comparison also the corresponding curve of Figure 1 (computed on the spelling list alone) is reproduced.

5. Conclusions

For solving the problem of recognizing huge name vocabularies for automating large scale directory assistance a systematic statistical combination of all available information items was studied.

Configuring the recognizer's active lexicon in a hierarchical manner and employing spelling of the last name as an entrance step avoids the usage of large (simultaneously active) vocabularies. Therefore, applying this method prevents pruning problems and allows the creation of real-time systems.

The application of newly proposed confidence measures for N-best lists allows the early detection of misunderstood turns.

Applying these techniques, acceptable automation rates can even be achieved for the largest cities. Our first, not yet optimized results show an automation rate of e.g. 72% of the inquiries for a database of 1.3 million entries with a remaining error rate of only 6% (or 62% with an error rate of 2%).

6. REFERENCES

1. 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.
2. B. Kaspar, G. Fries, K. Schuhmacher, and A. Wirth, "FAUST – A Directory-Assistance Demonstrator", *Proc. EUROSPEECH*, pp. 1161–1164, Madrid, Sept. 1995.
3. C. A. Kamm, K.-M. Yang, C. R. Shamieh, and S. Singhal, "Speech Recognition Issues For Directory Assistance Applications", *Proc. IVTTA*, vol. 1, pp. 15–19, Kyoto, Japan, Sep. 26–27 1994.
4. H. Hild and A. Waibel, "Recognition of Spelled Names over the Telephone", *Proc. ICSLP*, vol. 1, pp. 346–349, Philadelphia, PA, Oct. 1996.
5. F. Seide and A. Kellner, "Towards an Automated Directory Information System", *Proc. EUROSPEECH*, vol. 3, pp. 1327–1330, Rhodes, Greece, Sept. 1997.
6. A. Kellner, B. Rueber, and H. Schramm, "Strategies for Name Recognition In Automatic Directory Assistance Systems", *Proc. IVTTA, to appear*, Torino, Italy, Sept. 1998.
7. M. Weintraub, "LVCSR Log-Likelihood Ratio Scoring for Keyword Spotting", *Proc. ICASSP*, vol. I, pp. 297–300, Detroit, MI, May 1995.
8. B. Rueber, "Obtaining Confidence Measures from Sentence Probabilities", *Proc. EUROSPEECH*, vol. 2, pp. 739–742, Rhodes, Greece, Sept. 1997.