ELSEVIER

Speech Communication 17 (1995) 19–38

SPEECH COMMUNICATION

Continuous speech dictation – From theory to practice [☆]

V. Steinbiss [a,*], H. Ney [a,1], U. Essen [a], B.-H. Tran [a], X. Aubert [a], C. Dugast [a], R. Kneser [a], H.-G. Meier [a], M. Oerder [a], R. Haeb-Umbach [a], D. Geller [a], W. Höllerbauer [b], H. Bartosik [b]

[a] *Philips GmbH Forschungslaboratorien, 52066 Aachen, Germany*
[b] *Philips Dictation Systems, 1102 Wien, Austria*

Received 13 July 1994; revised 22 March 1995

Abstract

This paper gives an overview of the Philips research system for phoneme-based, large-vocabulary, continuous-speech recognition. The system has been successfully applied to various tasks in the German and (American) English languages, ranging from small vocabulary tasks to very large vocabulary tasks. Here, we concentrate on continuous-speech recognition for dictation in real applications, the dictation of legal reports and radiology reports in German. We describe this task and report on experimental results. We also describe a commercial PC-based dictation system which includes a PC implementation of our scientific recognition prototype. In order to allow for a comparison with the performance of other systems, a section with an evaluation on the standard Wall Street Journal task (dictation of American English newspaper text) is supplied. The recognition architecture is based on an integrated statistical approach. We describe the characteristic features of the system as opposed to other systems: 1. the Viterbi criterion is consistently applied both in training and testing; 2. continuous mixture densities are used without tying or smoothing; 3. time-synchronous beam search in connection with a phoneme look-ahead is applied to a tree-organized lexicon.

Zusammenfassung

Dieser Artikel gibt einen Überblick über den phonembasierten Philips-Spracherkenner für fließend gesprochene Spracheingabe mit großem Erkennungsvokabular. Das System wurde erfolgreich in mehreren Anwendungen in deutscher und (amerikanisch) englischer Sprache eingesetzt. Die Anwendungen reichten von kleinem zu sehr großem Vokabular. In diesem Artikel beschränken wir uns auf fließend gesprochene Sprache in einer echten Diktieranwendung (Rechtsanwaltsbüro und Röntgenabteilung, beides in Deutsch). Neben der Beschreibung der Anwendung und der Auswertung von Versuchsergebnissen gehen wir auch auf ein kommerziell erhältliches PC-basiertes System ein, das auf unserem Forschungsprototypen basiert. Um einen Vergleich unseres Prototypen

[☆] This paper is based on a communication presented at the ESCA Conference EUROSPEECH-93 and has been recommended by the EUROSPEECH-93 scientific program committee.

[*] Corresponding author. E-mail: tran@pfa.philips.de.

[1] Now Lehrstuhl für Informatik VI, University of Aachen (RWTH), D-52056 Aachen, Germany. E-mail: ney@informatik.rwth-aachen.de.

SSDI 0167-6393(95)00012-7

mit anderen Systemen zu erleichtern, ist ein Abschnitt über unsere Ergebnisse beim Wall-Street-Journal-Test des ARPA (gelesene Zeitungstexte in amerikanischem Englisch) enthalten. Die Systemarchitektur basiert auf einem integrierten statistischen Ansatz. Wir legen bei der Darstellung einen Schwerpunkt auf die Aspekte, in denen sich unser System von anderen stärker unterscheidet: 1. der Viterbi-ansatz wird konsistent sowohl beim Training als auch beim Testen verfolgt; 2. wir verwenden kontinuierliche Mischverteilungen ohne "Tying" oder Glättung; 3. bei der Suche verwenden wir zeitsynchrone Breitensuche in Verbindung mit einer schnellen Vorausschau auf Phonemebene ("fast look-ahead") sowie ein als Baum organisiertes Aussprachelexikon.

Résumé

Cet article présente une vue générale du système de reconnaissance de parole continue pour de grands vocabulaires développé par les chercheurs de Philips. Utilisant des modèles de phonèmes, ce système a été appliqué avec succès à diverses tâches couvrant l'éventail des petits jusqu'aux très grands vocabulaires, dans les langues allemande et anglo-américaine. Ce texte est consacré aux applications de la reconnaissance de la parole à des tâches de dictée réelle, en particulier celles concernant des rapports juridiques et radiologiques dans la langue Allemande. Ces tâches sont décrites de même que les résultats obtenus expérimentalement. Nous décrivons également une version commerciale d'un système de dictée issue de notre prototype et qui a été implémentée sur PC. Afin de rendre possible une comparaison des performances avec d'autres systèmes, une section est consacrée à l'expérimentation sur des données provenant du quotidien Américain "Wall Street Journal", incluant les tests réalisés lors de la procédure d'évaluation de novembre 1993. L'architecture générale est fondée sur une approche statistique intégrée. Par rapport à d'autres systèmes, les caractéristiques majeures qui se dégagent de notre système sont: 1. l'application constante du critère de Viterbi aussi bien pour l'apprentissage que pour la reconnaissance; 2. l'usage de mixtures de densités de probabilité continues sans recourir à aucune forme de partage ou de lissage; 3. un décodage synchrone avec une technique d'élagage en faisceau utilisée conjointement avec une organisation en arbre du lexique et une méthode d'anticipation rapide du phonème suivant.

Keywords: Continuous speech recognition; Large vocabulary recognition; Acoustic model; Hidden Markov model (HMM); Language model; Search; Dictation

1. Introduction

For large-vocabulary, continuous-speech recognition, there are a number of operational prototype systems in research, some of them participating in the ARPA[2] research programme or its evaluations. For the recognition of isolated word input of around 30K (30 000) words, there are commercial systems available from IBM and Dragon Systems. Like the above mentioned systems, the prototype system described in this paper is based on techniques of statistical pattern recognition and stochastic modelling, where training data are heavily exploited and local decisions are avoided as far as possible. See (Ney, 1993b; Ney et al., 1994a) for references.

The characteristic features of the approach to be presented are:
- A large-size acoustic vector capturing first and second-order derivatives is used. There is no splitting into separate streams as in most other systems that use tied mixtures.
- The Viterbi criterion is used both in training and recognition. Continuous mixture densities are used in a way that amounts to what could be called 'statistical template matching'.
- Linear discriminant analysis (LDA) improves the acoustic analysis.
- For bigram language modelling, a non-linear interpolation has been developed that gives consistently lower perplexities than linear interpolation, especially for small training corpora.
- The concept of time-synchronous beam search has been extended towards a tree organization of the pronunciation lexicon so that the search

[2] Advanced Research Projects Agency (U.S.-American organization funding, among others, speech recognition and understanding research)

effort is significantly reduced. A phoneme look-ahead technique results in an additional improvement. A PC based implementation (cf. Section 9) underlines the efficiency of this search strategy.

The organization of the paper is as follows. We first summarize the statistical approach to speech recognition and the experimental conditions of our dictation task. We then describe the four main entities of our system: acoustic analysis, acoustic-phonetic modelling, language modelling and search; experiments are included within the sections. To allow a comparison with the performance of other systems, a section on our Wall Street Journal system including the November 1993 evaluation (dictation benchmark test, American English) is supplied. The final section describes a PC based implementation of our system.

2. System architecture

Fig. 1 presents a block diagram of the system architecture. In the pre-processing step of *acoustic analysis*, the speech signal is transformed into a sequence of acoustic vectors $x_1,...,x_T$ (over time $t = 1,...,T$). As the speech signal, and thus this sequence of observations, is not exactly reproducible, a statistical approach is used to model its generation. Statistical decision theory tells that in order to minimize the probability of recognition errors, one should decide for the word sequence $W = w_1,...,w_N$ (of unknown length N) that maximizes (Jelinek, 1976; Jelinek et al., 1992)

$$\Pr(w_1,...,w_N|x_1,...,x_T) = \frac{\Pr(x_1,...,x_T|w_1,...,w_N)\Pr(w_1,...,w_N)}{\Pr(x_1,...,x_T)}.$$

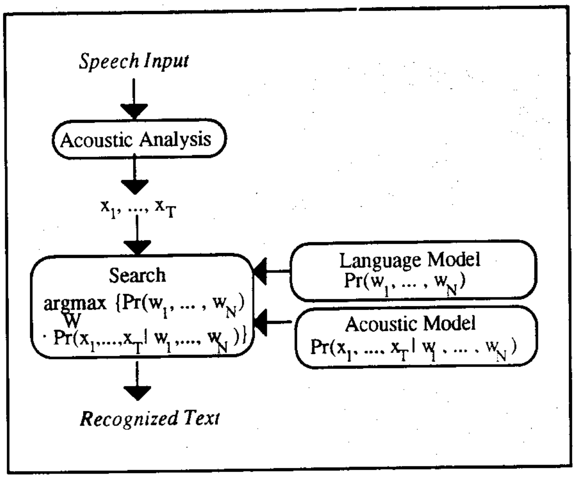
Speech Input
Acoustic Analysis
$x_1, ..., x_T$
Search
$\underset{W}{\text{argmax}} \{\Pr(w_1, ..., w_N) \cdot \Pr(x_1,...,x_T| w_1,..., w_N)\}$
Language Model
$\Pr(w_1, ..., w_N)$
Acoustic Model
$\Pr(x_1, ..., x_T| w_1, ..., w_N)$
Recognized Text

Fig. 1. System architecture.

As the denominator is constant for a given observation, this amounts to finding $w_1,...,w_N$ that maximizes

$$\Pr(w_1,...,w_N)\Pr(x_1,...,x_T|w_1,...,w_N). \qquad (1)$$

The first term, the a priori probability of word sequences $\Pr(w_1,...,w_N)$, is independent of the acoustic observations and is completely specified by the *language model*. It reflects the system's knowledge of how to concatenate words of the vocabulary to form whole sentences and thus captures syntactic and semantic restrictions.

The *acoustic-phonetic modelling* is reflected by the second term. $\Pr(x_1,...,x_T|w_1,...,w_N)$ is the conditional probability of observing the acoustic vectors $x_1,...,x_T$ when the words $w_1,...,w_N$ were uttered. These probabilities are estimated during the training phase of the recognition system. A large-vocabulary system typically is based on subword units like phonemes, which are concatenated according to the *pronunciation dictionary* to form word models.

The decision on the spoken words must be taken by an optimization procedure which combines information of the language model and of the acoustic model, the latter being based on the phoneme models and the pronunciation dictionary. The optimization procedure is usually referred to as *search* in a state space defined by the knowledge sources.

3. Experimental conditions

Before we focus on our dictation task, let us briefly describe the other conditions under which our speech recognition system is used. While it remains essentially the same system, several obvious modifications reflect the varying needs of these tasks. Giving two obvious examples, we use ("soft") m-gram language models for dictation

Table 2
Effect of LDA on the word-error rate (in %). About 3 hours of training material, 4 000 densities (cf. also Table 9)

Speaker	no LDA	LDA
M-60	12.3	10.4
M-61	15.0	12.3

tion for improving the discrimination between classes in a high-dimensional vector space (Duda and Hart, 1973, pp. 114 ff.). The basic idea is to find a linear transformation such that a suitable criterion of class separability is maximized. The transformation is obtained as the eigenvector decomposition of the product of two scatter or covariance matrices, the total-scatter matrix and the inverse of the average within-class scatter matrix. Recently, this technique has been successfully applied to speech recognition, for both small (Hunt and Lefebvre, 1989; Haeb-Umbach et al., 1993) and large-vocabulary tasks (Haeb-Umbach and Ney, 1992).

When applying LDA to speech recognition, the choice of the proper classes to be discriminated is not obvious – are they whole phonemes, phoneme states or the mixture components of a state? Our experiments indicated that the states are a good choice. The computation of the LDA transform is further complicated by the time alignment problem. Therefore, we use a three-step training. With our standard iterative training we obtain a segmentation of the training data, which provides the class labels for the subsequent estimation of the LDA transform. The third step is a new iterative training using LDA-transformed acoustic vectors.

Table 2 shows the improvement by LDA. Note that since a *single* class-*in*dependent transformation matrix is used, the matrix multiplication is done in the acoustic front end once per frame rather than for each log-likelihood calculation. Even for speaker-independent recognition, one single transformation gives satisfactory results.

5. Acoustic-phonetic modelling

The acoustic conditional probabilities $\Pr(x_1,...,x_T|w_1,...,w_N)$ are obtained by concatenating the corresponding word models, which again are obtained by concatenating phoneme models according to the pronunciation lexicon. We use inventories of 40–50 phoneme symbols including symbols for silence and maybe glottal stop. (For the English language, triphones are used as basic units; cf. Section 8.2.) As in many other systems, these subword units are modeled by stochastic finite-state automata, the so-called Hidden Markov Models (HMMs) (Baker, 1975; Jelinek, 1976; Levinson et al., 1983).

For each state s of the HMM, there is an emission probability density $q(x_t|s)$ of generating the vector x_t. The phoneme unit shown in Fig. 2 has a tripartite structure in order to take account of left and right acoustic dependences. Each of the three parts consists of two states with identical emission distributions. The transition probabilities, which allow loop, jump and skip, are tied over all states. Unlike most other HMM structures, this structure has a simple duration model whose most likely duration of 60 ms is close to the average phoneme duration.

No pronunciation variants are used in the pronunciation lexicon, such that the emission distributions have to model deviations from the standard pronunciation as well as coarticulatory effects. The best results were obtained for continuous mixture densities

$$q(x_t|s) = \sum_k c_k(s) b_k(x_t|s),$$

$$\text{with } 0 \leqslant c_k(s) \leqslant 1 \text{ and } \sum_k c_k(s) = 1, \qquad (3)$$

where the so-called component densities $b_k(\cdot|\cdot)$ are unimodal densities such as Gaussians or (as

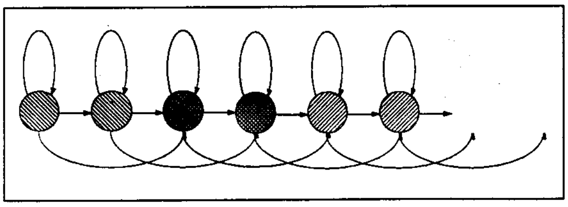

Fig. 2. Topology of phoneme HMM.

Table 3
Error-rate as a function of training set size and number of densities. Speaker M-60, vocabulary size 12 073 words, test-set perplexity 113

Training material	0.7 h	1.2 h	2.0 h	3.2 h	9.5 h
No. of densities					
4 000	16.1%	14.4%	13.1%	12.3%	11.4%
8 000	—	13.4%	12.9%	11.7%	10.8%
16 000	—	—	—	11.6%	10.1%
32 000	—	—	—	—	10.0%
64 000	—	—	—	—	9.1%

in our system) Laplacians:

$$b_k(x_t|s) = \prod_n \left(\frac{1}{2\nu_n} \right) \times \exp\left(- \sum_n \frac{|x_t(n) - r_{k,s}(n)|}{\nu_n} \right). \quad (4)$$

n is the index of the vector components. Each density is completely specified by its location vector $r_{k,s}$. The vector of absolute deviations, $(\nu_1,...,\nu_N)^T$, is assumed to be independent of both the component densities and the states and thus serves as an overall scaling for the acoustic vectors.

In contrast to other systems, the Viterbi criterion is used both in training and recognition. This applies even to the level of mixture components, such that the sum over the component densities in Eq. (3) is replaced by their maximum (Ney, 1993b).

Table 3 shows how the error-rate [3] depends on the training-set size and the acoustic resolution. Monophones (i.e. context-independent phonemes) were used here; we expect improvements with context-dependent phonemes.

While we typically develop our system on the speaker-dependent German dictation task, we also successfully benchmarked our system on both the speaker-dependent and the speaker-independent part of the well-known American English DARPA (Defense Advanced Research Projects Agency) RM (resource management) task (Aubert et al., 1993; Ney, 1993b) and on the ARPA Wall Street Journal (WSJ) task. The major modifications of our system and the WSJ benchmark results are described in Section 8.

6. Language modelling

The language model provides, for each word sequence, an estimate of probabilities $\Pr(w_1,...,w_n)$ usually expressed by m-gram models (cf. below) which have established themselves as both a good way to reliably estimate the parameters and to keep them limited so they can be stored and retrieved. In view of the sizes of available corpora, we typically use word bigram models or category-based bigram models (bigram class models) with automatically generated classes (Kneser and Ney, 1993). An overview about more general techniques in language modelling can be found in (Ney et al., 1994b).

While maximum-likelihood estimation would suggest to take relative frequencies of bigram counts, it is common knowledge that these are particularly bad estimates and that smoothing is important. The smoothing method that we use is different from those used in other systems and is explained in the following section in more detail. With this method, we achieve better results than with backing-off (Katz, 1987) or linear interpolation.

6.1. Stochastic bigram and trigram models

The task of providing probabilities $\Pr(w_1,...,w_n) > 0$ is usually reduced to the problem of estimating conditional probabilities $\Pr(w_j|w_1,...,w_{j-1})$ with given history $w_1,...,w_{j-1}$ which determines the joint probabilities by the product

$$\Pr(w_1,...,w_n) = \prod_{j=1}^{n} \Pr(w_j|w_1,...,w_{j-1}).$$

Because of the limited training data one has to share the same distribution for different histories, e.g. histories which coincide in the last $m-1$

[3] As usual, the word error-rate is defined as the ratio *(deletions + insertions + substitutions)/spoken words*. In contrast to word accuracy, defined as *correctly recognized words / spoken words*, erroneously inserted words count as errors.

positions. Depending on the amount and structure of training data we typically use only m-gram models with $m = 2$ (bigram) or $m = 3$ (trigram). Even for such small history lengths, there are a lot of possible bi- or trigram events which have not been observed during training before. So we are faced with the problem of guessing a positive probability for an event which has never been observed before. For this we have to use further knowledge about the stochastic process we want to describe.

Beside the well known technique of linear interpolation, the theory of most of the commonly used estimators was established in 1953 by Good (1953) who worked out an idea of Alan M. Turing. But in order to come up with practically useful 'Turing–Good' estimators one has to use some kind of smoothing.

The non-linear interpolation scheme used in our system has the advantage to do this in a way which is easy to implement. More precisely, in case of bigram and trigram models (Meier and Ney, 1994), it is possible to make a first-order approximation of the Turing–Good formula which simplifies it to subtracting a constant d (typically between zero and one) from counts greater than d. Redistributing the gained probability mass to some a priori distribution q leads to the concept of non-linear interpolation as introduced by (Ney and Essen, 1991).

To be more explicit, e.g. for a bigram application, let us denote the count of some bigram (ν,w) in a given training corpus by $N(\nu,w)$. Then we may define the estimator for a bigram language model by

$$p(w|\nu) := \begin{cases} \dfrac{N(\nu,w) - d + \beta_\nu q(w)}{N(\nu)} \\ \quad \text{if } N(\nu,w) > d, \\ \dfrac{\beta_\nu}{N(\nu)} q(w) \\ \quad \text{if } N(\nu,w) \le d, \end{cases}$$

if $N(\nu) := \Sigma_w N(\nu,w)$ is assumed to be positive and β_ν is chosen to assure the constraint $\Sigma_w p(w|\upsilon) = 1$. Here q is usually chosen to be a unigram distribution. Defining a discounting function $\delta(\nu,w) := \min\{d, N(\nu,w)\}$ we easily get $\beta_\nu = \Sigma_w \delta(\nu,w)$ as well as

$$p(w|\nu) = \frac{N(\nu,w) - \delta(\nu,w) + \beta_\nu q(w)}{N(\nu)},$$

which describes a general interpolation scheme between q and the relative frequency distribution. The name 'non-linear interpolation' indicates the difference to the well-known 'linear interpolation' with parameter α which precisely appears if we choose $\delta(\nu,w) := \alpha N(\nu,w)$.

6.2. Application-specific experimental results

From the theoretical derivation it is clear that non-linear interpolation is designed to incorporate different statistical knowledge (e.g. about unigram and bigram) in a way which respects the advantage of the Turing–Good estimator of providing better estimates even with relatively small training data.

In fact, in practice there are typically only small training corpora available which reflect the application and the speaker-specific characteristics. To compare the performance of non-linear interpolation and linear interpolation, we took spoken sentences from two lawyers (M-60 and M-61) and two radiologists (M-72 and M-73) as well as a larger corpus of written radiology reports (REP; see Table 4) to calculate the different test-set perplexities. (Recall that the logarithm of perplexity can be viewed as the empirical entropy for the actual test set.) As seen in Table 6, in all cases non-linear interpolation yields significantly lower (i.e. better) perplexities than linear interpolation. Furthermore, the relative gain becomes smaller for larger training material.

Table 4
Data sizes in words for specific applications

Data	Test	LM training	Lexicon
M-60	2 781	61 130	12 073
M-61	3 039	71 208	15 188
M-72	2 095	50 192	13 095
M-73	2 296	54 375	13 095
REP	569 767	915 858	40 630

Table 5
Test-set perplexity for *m*-gram language models ($m = 1,2,3$)

Non-linear	Unigram	Bigram	Trigram
M-60	818.9	112.2	81.2
M-61	933.4	183.2	151.2
M-72	1065.9	286.9	264.3
M-73	531.8	41.9	30.7
REP	832.4	91.9	66.5

Table 6
Test-set perplexity for different discounting methods

Bigram-LM	Linear	Non-linear
M-60	127.0	112.2
M-61	206.4	183.2
M-72	299.8	286.9
M-73	47.4	41.9
REP	97.4	91.9

Table 7
Test-set perplexities when using only written training corpora ("REP"; without data as being dictated)

Test set	Unigram	Bigram
M-72	1822.5	705.2
M-73	1599.3	365.1

It should be noted that the unigram distribution q used in the experiments was also calculated with a non-linear interpolation scheme using a uniform distribution as background knowledge (see Table 5).

Of course all techniques may be applied also to trigrams using a conditional bigram distribution as the general background model. Even for small corpora it is possible to have a gain in perplexity if the training material gives a good coverage of frequently used phrases in a very special application (see Table 5).

To indicate that there is a great difference between specific well-tailored training material and general application-specific data, we used the unigram and bigram models trained on written radiology reports (REP) to calculate test-set perplexities on spoken radiology reports of M-72 and M-73. To make test results comparable with Table 5, we used the lexicon of the M-72/M-73 corpus.

Tables 7 and 5 show that the language models trained on a small-sized corpus of speaker specific sentences that were transcribed as spoken ("as-it-is files") perform much better than the models trained on the larger speaker indepen-

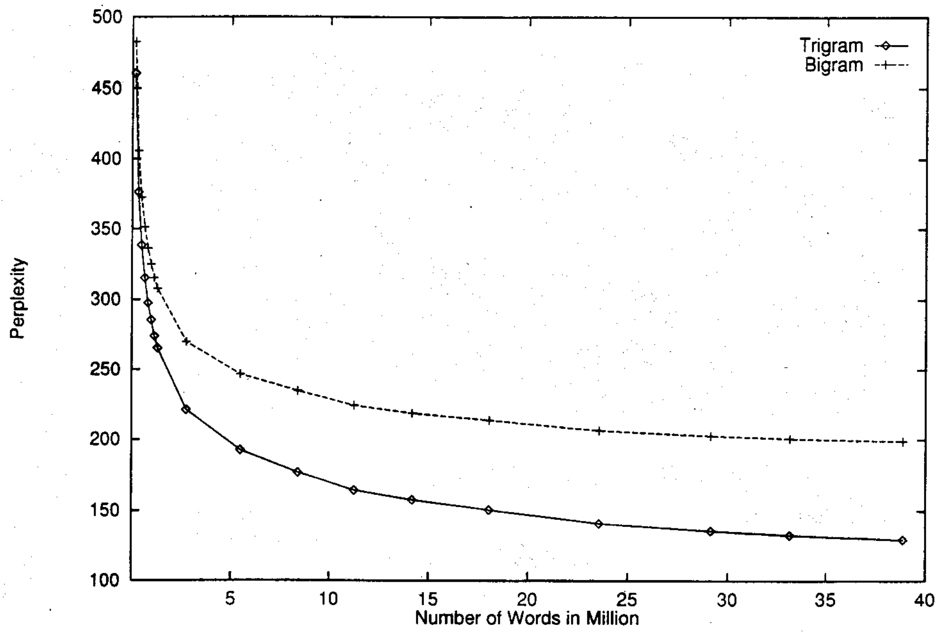

Fig. 3. Test-set perplexity for bigram and trigram LM depending on training set size.

dent written text. This seems to indicate that specific data material is more important than some general kind of knowledge. Another reason for this effect might be the general difference between spoken and written language. Most obvious examples for this difference (like abbreviations and punctuation) stem from some kind of mismatch between the words in written and spoken text.

6.3. *Perplexity gain for large corpora*

Although the techniques just presented perform quite well with small training material, there is still a strong gain in perplexity when using larger training corpora. To see the dependence between language model performance and training size we took different sized subcorpora of up to 39 million words from the well-known Wall Street Journal corpus.

Fig. 3 shows the significant loss in performance when only small corpora are used for training: the more (application specific) data, the better. This is even more true for a trigram model.

7. The search procedure

Time-synchronous beam search has successfully been used in the Philips continuous-speech recognizer for several years (Ney et al., 1992b). We found that it is efficient also for 10K or more words (Ney et al., 1992a). First, all knowledge sources are available at the same level in the integrated search. Second, all hypotheses refer to the same acoustic vector sequence in time-synchronous search. These two key points allow a drastic reduction of the actual search space by pruning less promising hypotheses.

Recently, we increased the vocabulary size in our WSJ benchmark system up to 45K words. Our positive experiences are described in Section 8.6.

7.1. *Tree lexicon*

A straight-forward approach of constructing the search space is to synthetically build up word models from concatenating the appropriate phoneme models as given by the pronunciation lexicon. In this space, different copies of the same phoneme occur due to the lexical constraints. For similar reasons, the language model restrictions make it necessary to introduce several copies of the same word, representing contexts that allow for different continuations. This organization, where each state belongs to exactly one word, will be called *linear lexicon*.

When the lexicon grows larger, e.g. from 1K to 10K words, it is more efficient to arrange the pronunciation lexicon as a tree of phonemes (*tree lexicon*). The compression factor for the tree lexicon as compared to the linear lexicon is even surpassed by the reduction in the number of active states, because most of the active states are located in the word beginnings (near the tree's root).

7.2. *Forest search*

The tree organization of the lexicon also has an undesired consequence for the organization of the search space. In contrast to a linear lexicon, the word identities are unknown at the word beginnings. Particularly for a bigram language model, this means that separate tree copies have to be held, depending on the predecessor word. While the potential search space is blown up by a factor of the vocabulary size, e.g. 10K, the actual search space grows much more moderately, typically by only a factor of 2. The tree organization is thus very beneficial for large-vocabulary tasks. A detailed discussion with experiments is given in (Ney et al., 1992a).

7.3. *Phoneme look-ahead*

The phoneme look-ahead additionally reduces the number of active states by estimating whether a started phoneme will or will not survive the next few time frames (in our system typically 60 ms). In a first step, the likelihood of each phoneme ahead of the current time frame is estimated by carrying out a time-alignment. Then, each time a state hypothesis crosses a phoneme boundary, these figures are used as probability estimates for the best path extensions both of this and of any other state, which in turn are used to perform an additional pruning (Haeb-Umbach and Ney, 1991).

For the phoneme look-ahead, the original phoneme models are used without any simplification. Note that, in particular for the case of monophones, the number of generic states is much smaller than the number of state hypotheses. (Conversely, a non-modified application to a system with many triphones is not advantageous.) The likelihood scores are stored for later use in the detailed match. Like the conventional search, the look-ahead is sped up by beam pruning; in addition, there is no need for book-keeping as in the detailed match. To further reduce computation, the look-ahead is carried out only every other time frame. For the omitted time frame, the look-ahead scores of the previous time frame are used.

7.4. Peaks in the search space: histogram pruning

Conventional beam pruning uses a pre-specified constant threshold to specify the beam of active hypotheses: At each time frame, exactly the hypotheses with log-probabilities close enough to the optimum at that time remain active, i.e. are considered for expansion at the next time; the others are pruned.

When the pruning threshold is chosen to be large enough to avoid search errors, i.e. when the optimal path is only rarely being lost due to pruning, large peaks in the actual number of active hypotheses can occur. We frequently observed peaks of 1 or 2 million hypotheses and roughly 100 times larger than the average number of hypotheses, especially for non-speech sounds or corrupted speech.

We thus introduced an additional pruning criterion: a pre-specified upper limit on the number of active points. We called this *histogram pruning* because we use a histogram on the hypotheses' scores in order to determine a pruning threshold (below a given value) such that the number of active hypotheses remains always below a given maximal number of active hypotheses.

Quite astonishingly, the experiments indicated that it is possible to choose relatively small maximal numbers for the hypotheses without introducing search errors. 30 000 hypotheses maximum is a typical value for our dictation research prototype. Besides the significant reduction in peak storage size needed, there is a reduction in the average search costs of about 30%. A detailed description of the experiments is given elsewhere (Steinbiss et al., 1994).

7.5. Language-model (LM) look-ahead: smearing the expected LM probabilities over the tree

In the forest search organization for stochastic n-gram language models ($n > 1$), the potential search space consists of a large number of copies of the phonetic tree consisting of the recognition vocabulary. E.g., for a vocabulary of V words and a bigram LM, i.e. $n = 2$, there are $V^{n-1} = V$ copies of the phonetic tree. Informal experiments indicated that, due to beam pruning, the number of active hypotheses grows much smaller with n, like roughly a factor of 2 when going from unigram to bigram LM.

The word identities in the tree are only known at the word ends. Adding the LM log probabilities at the word ends leads to several effects that are disadvantageous for the search:

- As compared to linear search, the LM is employed with one word delay. But knowledge should be incorporated as early as possible.
- The scores of hypotheses change drastically when a word end is encountered. Especially, the pruning has to be larger than the largest LM score ("score" being defined in this paper as negative log probability).
- The same effect causes the examination of many useless word start hypotheses during silence after a word.

A remedy for all these pains is the incorporation of the LM scores as early as possible. For this purpose, in each search state, we introduce a new pruning criterion: instead of the usual score, we always investigate its sum with the minimum of the LM scores of all possible word continuations. A practical implementation and experimental results are described in (Steinbiss et al., 1994). We achieved reductions in search space by factors 3 to 5 with this method.

7.6. Longer span language models with lattice rescoring

7.6.1. Basic concept

During our first tests with forest search, we made informal experiments indicating that forest search works not only with a bigram LM but also with a trigram LM, with only moderate increase of the active search space by an additional factor of roughly 2. However, for the recognition with a trigram LM, we decided to choose a different approach with a search effort about the same as for a bigram LM. In this two-step approach, a word lattice is first generated with a bigram LM and subsequently rescored with a trigram LM. The approach is open to employ more complex LMs in this-post processing step.

7.6.2. Generation of the word lattice

A word lattice can be efficiently generated with only minor modifications of our time-synchronous beam search algorithm based on a tree lexicon. It essentially amounts to collecting the information about word-endings as they occur in the course of the left to right decoding process. This first pass simultaneously provides the best bigram-scored sentence hypothesis, the lattice overhead being virtually negligible in terms of CPU time.

As opposed to the word-graph generation technique presented in (Oerder and Ney, 1993), here we take full advantage of the bigram LM to constrain the lattice, without requiring any further optimization stage. More precisely, our analysis relies on the assumption that the position of a word boundary depends only on the word pair under consideration and not on further predecessor words. This simplification has been successfully used by BBN in their word-dependent *N*-Best algorithm (Schwartz and Austin, 1991) and is also known as the "word-pair approximation" (Ney, 1993a).

Therefore, in the present study the lattice is defined as a time-structured list of word hypotheses consisting of word identity, start and end times, acoustic score and predecessor word identity. It has to be stressed that the collection of word-ending information is done before the bigram LM recombination takes place, to preserve as much as possible different word sequences for subsequent use with a higher-order LM.

The computational complexity of this first pass is nearly identical to that of our bigram beam search, the efficiency of which having been further improved by the new handling of the LM probabilities (see Section 7.5).

7.6.3. Trigram rescoring of the lattice

In this second pass, the trigram language model is applied to the lattice at the phrase level. More precisely, the acoustic probabilities of the word hypotheses are combined with the trigram probabilities taking account of the predecessor-word as computed in the first pass. Searching for the optimal rescoring still proceeds time-synchronously and requires a Dynamic Programming (DP) recursion taking account of all time and predecessor constraints contained in the lattice (Ney, 1993a). The final output is the best trigram-scored sentence hypothesis under the lattice restrictions.

The optimality of this procedure (in the Viterbi sense) is preserved only under the following two conditions: the word-pair approximation for the position of a word boundary has to be valid and, next, the beam used for generating the lattice must be wide enough to keep enough phrase hypotheses for subsequent trigram rescoring.

In practice, this algorithm appears to work well with relatively modest lattice densities. The computational costs are quite small since this second pass does not require any further acoustic scoring at the state level. This follows from the word-pair assumption which implies that the word boundaries have already been optimized in the first pass.

Moreover, a careful list organization allows to achieve great efficiency (without requiring the cashing of the LM scores) to such an extent that the trigram rescoring represents only a few percent of the main bigram decoding CPU time.

8. Benchmarking our system on the Wall Street Journal task

8.1. Why and when benchmarking?

To some extent, comparison between speech recognizers is part of the scientific competition:

the quality of our work is largely reflected in the ability of the acoustic and language models to model reality – which is typically measured in terms of word error rate, given fixed experimental conditions. Moreover, reproducible benchmark tests allow us to validate importance and significance of improvements achieved by our colleagues and to check whether our system does what it ought to.

The boundary condition for system development differ somewhat from the development of a dictation system for real use. There is a lot of data available that well represents the task. As the only optimization criterion is performance in terms of error rate, we e.g. take a much finer acoustic resolution, and memory demands and processing time play a minor role here.

So far, we benchmarked our system on

- the TI digit string database (Haeb-Umbach et al., 1993);
- the DARPA RM (resource management) task (Ney, 1990; Aubert et al., 1993) and participated in the last official evaluation (Aubert et al., 1993);
- the November '92 and the November '93 evaluations (official participation for November '93) of the Wall Street Journal task (Aubert et al., 1994).

The latter is described in the subsequent section.

The ARPA WSJ corpus (Paul and Baker, 1992) consists (among others) of samples of read texts drawn from the Wall Street Journal publications and provides training and test material for SI continuous speech recognition in American English. Vocabulary sizes are typically ranging from 5K (closed) to 20K (open, i.e. there are out-of-vocabulary (OOV) words being uttered in the test sentences). In addition, standard bigram and trigram language models have been supplied by D. Paul from MIT Lincoln Lab.

8.2. System development

We first present some intermediate results illustrating the main development stages of our "WSJ systems". Unless specified, all (non-stressed) pronunciations were taken from the original Dragon lexicon, training was performed on the 84-speaker corpus, and recognition was done with a bigram LM. Experiments have been run on several WSJ0 development sets with non-verbalized punctuation and for various vocabularies (5K closed, 20K open and closed).

The first step shows the reduction of the error rate when using phone models that capture gradually more contextual dependencies.

Our interest for left-diphones stemmed from the fact that they preserve the lexical tree structure of monophone transcriptions as opposed to triphones. However triphones clearly lead to more accurate models. Therefore, next stages have been running with the set of 736 triphones occurring more than 150 times in the WSJ0 training script.

The second step concerns the effect of linear discriminant analysis (LDA) and of gender-dependent (GD) estimation that has been used both for the LDA transform and for the mixture parameters as well (Aubert et al., 1993). Our experiments are summarized in Table 9.

The best configuration is achieved with a single gender independent (GI) LDA transform followed by GD mixture estimation. Adding the uni-sex models to the male- and female-specific models only brings a further marginal improvement.

Table 8
Influence of contextual units (Dev-5K)

No. of units	Type	No. of densities	Error rate	Progress
43	Monophones	33 K	18.5%	±0% (Ref.)
772	Left-diphones	37 K	15.2%	−18%
772	Left-diphones	115 K	14.0%	−24%
736	Triphones	73 K	13.1%	−30%

Word-error rate = del + ins + sub.

The third step involves the LM rescoring technique in word lattices generated with a bigram LM and shows the error reduction when going from bigram to trigram.

When switching from bigram to trigram, both the test-set perplexity and the error rate are significantly reduced. It is interesting to observe that – at least for our closed-vocabulary experiments – the error rate decreases like the square root of the bigram-to-trigram perplexity ratio. Note however that this is nothing but a rule of thumb deduced from limited experimental data. For open vocabulary, the interpretation is complicated by the presence of out-of-vocabulary words that constitute about 2% of the test words and give rise to additional insertion errors. Moreover, in this case the perplexity measures are no longer that reliable.

Last, we give a few figures concerning some characteristic properties of the bigram search cost:

- When using triphones, the number of arcs in the first two generations of the tree lexicon are multiplied by respectively 6 and 2 with respect to the monophone tree. However, due to the improved precision of the triphone models, the average number of state hypotheses in the beam search is actually smaller!
- The improved LM-based pruning reduces the average number of hypotheses by a factor of 3 to 5 compared to the original handling of bigram scores.
- When the vocabulary grows from 5K to 20K words, the average number of hypotheses increases by not more than 50% due to the lexical tree.

Fig. 4 summarizes the development steps up to this stage.

8.3. Description of the evaluation system

Two systems have been set-up differing mainly in the number of triphones and the amount of training data. In each case, the mixture density parameters have been estimated gender-dependently with respectively male, female and uni-sex models. During decoding, the word sequence achieving the highest cumulated probability has been taken for the recognized sentence. Table 11 contains the main system characteristics.

Table 9
LDA and mixture gender dependent (GD) estimation (Dev-5K)

LDA?	Gender (LDA)	No. of densities	Gender (Dens.)	WER%	Progress
No	—	2∗65 K	M/F	12.4	±0% (Ref.)
Yes	M/F	2∗65 K	M/F	11.3	−9%
No	—	1∗133 K	GI	12.4	±0%
Yes	GI	1∗139 K	GI	10.6	−15%
Yes	GI	2∗123 K	M/F	9.7	−22%
Yes	GI	" + 139 K	M/F/all	9.4	−24%

Gi = gender-independent, over male (M) and female (F) speakers.

Table 10
From bigram to trigram language model

Corpus (mode)	Bigram		Trigram		Relative reduction
	WER %	Perp.	WER %	Perp.	
Dev 5K closed [a]	10.6	110	7.9	62	−25%
Dev 5K closed [b]	9.7	110	7.3	62	−24%
Dev 20K Closed [a]	18.8	242	15.1	155	−20%
Dev 20K open [a]	19.9	205	16.4	136	−16%

WER = word-error rate (del + ins + sub).
[a] and [b] refer respectively to GI and GD (M/F) mixture modeling both after GI LDA (cf. Table 9).

Table 11
Our two evaluation systems

Name	Training data	Training time	Lexicon	No. of monophones + No. of triphones
SI-84	WSJ0	ca. 15 h	LIMSI	45 + 740 tri
SI-284	WSJ0 + 1	ca. 80 h	Dragon	43 + 1864 tri

We used two lexica, which were provided by LIMSI and Dragon Systems, respectively. As indicated, the LIMSI lexicon has been used in the first system (trained over 84 speakers) while the Dragon lexicon has been used in the second one. LDA has been applied gender-independently based on 84 speakers (WSJ0 training data). The average number of Laplacian densities per state is about 45. The official bigram and trigram LM have been employed without any modification. These systems have been tested on the evaluation sets of November '92 and November '93 containing recordings from respectively 8 and 10 new out-of-training speakers.

8.4. Closed 5K lexicon

The two systems have been running on each 5K evaluation set with standard benchmark conditions, i.e. not using any side information about the utterances. Results for bigram and trigram LM are summarized in Table 12.

Concerning the LM influence, the error rates are approximately reduced like the square root of the perplexity ratio when going from bigram to trigram. This represents a recovery of 30% of the errors.

A clear improvement follows when more acoustic models are estimated using more training data. With respect to system SI-84, SI-284 achieves an improvement of about 20% on November '92 and 30% on November '93. This is attributed to the acoustically more difficult recordings of last evaluation data as might be inferred when considering the perplexities and the error rates of both sets.

8.5. Open 20K lexicon

Here we present the 20K results obtained with the second system SI-284 trained on 284 speakers.

When going from bigram to trigram, the 20K errors are now reduced by about 15%, i.e. somewhat less than could have been expected from the "square root of perplexity ratio" rule of thumb. However, the presence of about 2% of out-of-vocabulary words makes the analysis somewhat

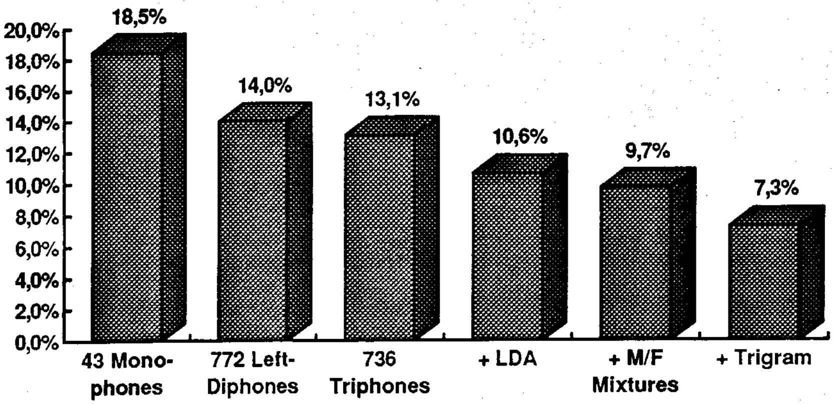

Fig. 4. Development steps from the baseline system with 43 monophones, no LDA, gender independent mixtures and bigram LM (perplexity 110) to the trigram (perplexity 62) system with LDA, triphones and gender dependent mixtures.

Table 12
Word-error rates on the 5K evaluation set

System	November '92		November '93	
	Bigram ($P = 111$)	Trigram ($P = 57$)	Bigram ($P = 106$)	Trigram ($P = 62$)
SI-84	(0.7/0.1) 7.4	(0.4/0.7) 5.0	(3.2/1.0) 12.7	(2.5/1.0) 9.4
SI-284	(0.5/0.8) 6.1	(0.4/0.5) 4.3	(2.8/0.8) 9.2	(1.8/0.6) 6.5

difficult as they give rise to additional insertion errors having some "cascade" effect on the trigram scores.

To conclude with these 5K–20K experiments, it might be observed that when using five times more acoustic training data together with a trigram LM, the overall gain in accuracy is close to 50% with respect to our baseline SI-84 bigram results, i.e. about one word error over two is recovered.

8.6. Extension to 45K lexicon

As mentioned in the preceding section, the test data used for the 20K evaluation contains out-of-vocabulary (OOV) words as it was actually collected from newspaper texts spanning a much larger vocabulary of 64K. The 20K lexicon was simply made of the 20K most frequent words measured over the LM training data consisting of about 39 million words. Hence, the WSJ corpus offers the possibility of studying the impact of vocabulary coverage on the recognition accuracy, an important aspect for any real-life dictation application.

In our experiments, we observe that a spoken word not in the lexicon causes typically more than one error (about 1.6 on average) due to the influence of neighbor words and the tendency of fragmenting a long unknown word into smaller ones. Thus there is an obvious interest for working with a drastically enlarged vocabulary. Table 14 gives the percentage of OOV words as a function of the vocabulary size for the two test-sets considered in the previous section.

By selecting the 45K most frequent words instead of 20K, the OOV rate decreases by about 1.5% which could potentially lead to an absolute error reduction of 2.5%.

However, for running this experiment a number of preliminary steps have to be fulfilled. First, the lexicon has to be extended with the phonetic transcriptions of the 25K added words and second, a new trigram language model has to be set up for this enlarged vocabulary (all our WSJ tests were made so far with the official 5K and 20K language models provided by MIT Lincoln Lab).

The first problem has been solved by using a Grapheme-to-Phoneme conversion system (Besling, 1994) which automatically generated all missing transcriptions in the Dragon 20K lexicon. For the estimation of a 45K trigram LM, the training data consisted of normalized texts taken from the Wall Street Journal publications in the period 1987–1989, providing a total of 38.9 million words covered by a 173K vocabulary (Paul and Baker, 1992). This led to a LM with 4.2 million bigrams and 15.8 million trigrams.

Table 15 gives the test-set perplexities obtained respectively with the official 20K and the 'home-made' 45K language models. For these figures, the OOV words appearing in the test sentences have been mapped to a single class of

Table 13
Evaluation results for 20K open lexicon

20K November '92 WER %		20K November '93 WER %	
Bi ($P = 205$)	Tri ($P = 139$)	Bi ($P = 221$)	Tri ($P = 143$)
(1.1/2.4) 14.0	(1.0/2.1) 11.9	(2.8/1.7) 17.3	(2.5/1.3) 14.9

WER given as (del/ins) tot = del + ins + sub.
P = Test-set perplexity for bigram and trigram.

Table 14
Frequency of OOV words versus vocabulary size

Test set	20K	45K	65K
November '92	1.9%	0.34%	0.0%
November '93	1.7%	0.35%	0.1%

Table 15
Test-set perplexities versus vocabulary size

Test set	20K Bigram/trigram	45K Bigram/trigram
November '92	205/139	219/146
November '93	221/143	233/146

Table 16
Trigram recognition results versus vocabulary size

Test set		20K	45K	Difference
November '92	OOV	1.9%	0.34%	−1.56%
	WER	11.9%	9.8%	−2.1% (18% rel.)
November '93	OOV	1.7%	0.35%	−1.35%
	WER	14.8%	13.3%	−1.5% (10% rel.)

words grouping all unknown words met in the LM training corpus. So, due to the treatment of the OOV words, these perplexity measures are only lower bounds. When the vocabulary is enlarged, unfrequent words that were OOV are now included and tend to be of low probability, which explains the increase of perplexity when going from 20K to 45K.

Recognition has been performed with gender-dependent (male/female) acoustic models of 1864 triphones estimated from 284 training speakers and the two-step decoding strategy has been applied: first a word lattice has been generated using the bigram tree search algorithm and next the lattice was searched for the best sentence using the trigram LM. The recognition results are summarized in Table 16.

The improvement is particularly striking in the first test-set where most of the errors due to 'OOV' words could be recovered while the gain is still appreciable for the acoustically more difficult recordings of the second test-set. In both cases, a significant decrease of the Word-Error Rate (WER) is achieved just by giving the recognizer a larger number of possible word candidates.

Concerning the search effort, the overall decoding cost is increased by not more than 15% when going from 20K to 45K. This follows from the tree-organization of the lexicon as illustrated in Table 17.

Although the total number of arcs in the tree increases in the same proportion as the vocabulary size, the numbers of arcs in the first two generations (G1 and G2) exhibit a much smaller increase and these arcs are precisely responsible for most of the search effort when using a time-synchronous beam-pruning strategy. Hence, the highly valuable property that the decoding cost increases much slower than the actual vocabulary size.

9. A PC based continuous-speech recognition system for dictation

Real-world dictation, which is typically connected with large and open vocabulary, is a difficult task that pushes today's technology to its limits. Despite the considerable progress made in recent years, even for co-operative speakers and restricted domains like free-text medical reporting, error-free speech recognition so far cannot be achieved.

Up to now, large-vocabulary systems for dictation have required isolated word input. While this reduces both the word-error rate and computational costs as compared to continuous-speech input, it burdens the user with an unnatural speaking style. In addition, dictating with pauses between words takes more time.

For people who professionally generate large amounts of texts, e.g. physicians and lawyers, text generation is characterized by a two-step process:

Table 17
Number of arcs in tree lexicon (G_i = number of arcs in generation i)

Voc.	Total no. of arcs	$G1$	$G2$	$G3$	$G4$	No. of homophones
20K	55728	350	1530	6586	10655	932/19980 = 4.7%
45K	124783	355	1904	10439	21523	3632/44979 = 8.1%
Ratio: 2.25	2.24	1.01	1.24	1.59	2.02	3.9

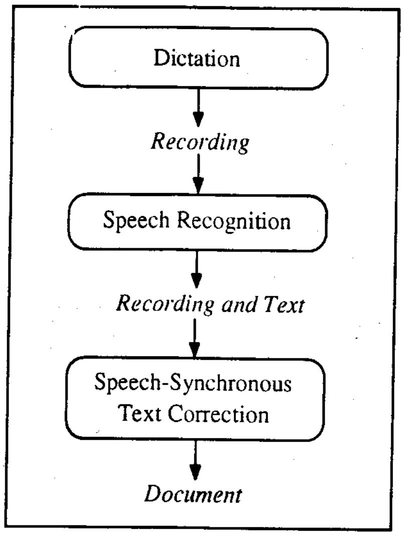

Fig. 5. System architecture of the PC based continuous-speech recognition system for dictation.

the phase of dictation, where speech is recorded either digitally or on tape, and a subsequent separate transcription phase where secretaries transcribe the dictations. (We ignore the proofreading in this discussion.)

The system developed at Philips Dictation Systems, Vienna, and described here adopts this non-interactive approach and thus allows the person to dictate with a natural speaking style and without an undesired distraction by the need of controlling a PC screen. After the speech is processed by the speech recognizer, the secretary has only to correct the recognition errors, which is both faster and a more interesting job to do.

This three-step approach naturally results in the system architecture as summarized in Fig. 5.

- The dictation is recorded using a microphone; the usual record/replay/fast-forward/rewind functionality is available. So, no change of work methodology is required. Punctuation should be verbalized. The recorded speech is stored on a fileserver in a PC network.
- Speech recognition runs remotely on a PC which is connected to the network. (A typical configuration currently used has a 486 processor and a 66 MHz clock rate.) An acoustic front-end performs the acoustic analysis. Recognition is made faster by a dedicated coprocessor board containing application-specific ICs. Depending on the speaker and the specific boundary conditions, recognition with a 10K–20K-word vocabulary runs in 1–3 times real-time.
- In contrast to typing the whole text, the secretary only corrects the errors that occurred in the recognition process. With a special speech-synchronous editor that uses the link between the recording and the text as given by the hypothesized word boundaries, it is possible to listen to parts of the recording while moving through the text.

Three measures have been taken to achieve the lowest possible error-rates without hindering the person who dictates:

- The system has been set-up in speaker-dependent mode such that each of the speakers gets optimal performance.
- Training starts with reading a specified text and is continued with the dictations that are being produced anyway, together with their proper transcriptions. After several hours of dictations, the system reaches optimal performance.
- A high-quality acoustic analysis together with a large number of mixture components guarantee a high acoustic resolution.

The first release of this system is a German version. Field trials are being carried out in several hospitals in Austria and Germany. The system was first shown to the public on the ECR (European Congress of Radiology) in Vienna in September 1993; an American English prototype version was presented on the RSNA (Radiological Society of North America) conference in Chicago in December 1993.

References

X. Aubert, R. Haeb-Umbach and H. Ney (1993), "Continuous mixture densities and linear discriminant analysis for improved context-dependent acoustic models", *Proc. Inter-*

nat. Conf. Acoust. Speech Signal Process., Minneapolis, MN, April 1993, pp. II-648-651.

X. Aubert, C. Dugast, H. Ney and V. Steinbiss (1994), "Large vocabulary continuous speech recognition of Wall Street Journal data", *Proc. Internat. Conf. Acoust. Speech Signal Process.'94, Adelaide,* Vol. II, pp. 129–132.

H. Aust, M. Oerder and F. Seide (1994a), "Experience with the Philips automatic train timetable information system", *Proc. IVTTA'94 Second IEEE Workshop on Interactive Voice Technology for Telecommunications Applications, Kyoto,* pp. 67–72.

H. Aust, M. Oerder and V. Steinbiss (1994b), "Database query generation from spoken sentences", *Proc. IVTTA'94 Second IEEE Workshop on Interactive Voice Technology for Telecommunications Applications, Kyoto,* pp. 141–144.

J.K. Baker (1975), "Stochastic modeling for automatic speech understanding", in *Speech Recognition,* ed. by D.R. Reddy (Academic Press, New York), pp. 512–542.

S. Besling (1994), "Heuristical and statistical methods for grapheme-to-phoneme conversion", in *Proc. KONVENS 1994,* ed. by H. Trost (Springer, Vienna), pp. 23–31.

S. Dobler, D. Geller, R. Haeb-Umbach, P. Meyer, H. Ney and H.-W. Ruehl (1993), "Design and use of speech recognition algorithms for a mobile radio telephone", *Speech Communication,* Vol. 12, No. 3, July, pp. 221–229.

R.O. Duda and P.E. Hart (1973), *Pattern Classification and Scene Analysis* (Wiley, New York).

I.J. Good (1953), "The population frequencies of species and the estimation of population parameters", *Biometrica,* Vol. 40, December, pp. 237–264.

R. Haeb-Umbach and H. Ney (1991), "A look-ahead search technique for large-vocabulary continuous-speech recognition", *Proc. Europ. Conf. on Speech Communication and Technology, Genova, September 1991,* pp. 495–498.

R. Haeb-Umbach and H. Ney (1992), "Linear discriminant analysis for improved large vocabulary continuous speech recognition", *Proc. Internat. Conf. Acoust. Speech Signal Process., San Francisco, CA, March 1992,* pp. I-13-16.

R. Haeb-Umbach, D. Geller and H. Ney (1993), "Improvements in connected digit recognition using linear discriminant analysis and mixture densities", *Proc. Internat. Conf. Acoust. Speech Signal Process., Minneapolis, MN, April 1993,* pp. II-239-242.

M.J. Hunt and C. Lefebvre (1989), "A comparison of several acoustic representations for speech recognition with degraded and undegraded speech", *Proc. Internat. Conf. Acoust. Speech Signal Process., Glasgow, May 1989,* pp. 262–265.

F. Jelinek (1976), "Continuous speech recognition by statistical methods", *Proc. IEEE,* Vol. 64, No. 10, April, pp. 532–556.

F. Jelinek, R.L. Mercer and S. Roukos (1992), "Principles of lexical language modeling for speech recognition", in *Advances in Speech Signal Processing,* ed. by S. Furui and M.M. Sondhi (Marcel Dekker, New York), pp. 651–699.

S. Katz (1987), "Estimation of probabilities from sparse data for the language model component of a speech recognizer", *IEEE Trans. Acoust. Speech Signal Process.,* Vol. 35, No. 3, March, pp. 400–401.

R. Kneser and H. Ney (1993), "Improved clustering techniques for class-based statistical language modelling", *Proc. Europ. Conf. on Speech Communication and Technology, Berlin, September 1993,* pp. 973–976.

S.E. Levinson, L.R. Rabiner and M.M. Sondhi (1983), "An introduction to the application of the theory of probabilistic functions of a Markov process to automatic speech recognition", *Bell Syst. Tech. J.,* Vol. 62, No. 4, April 1983, pp. 1035–1074.

H.-G. Meier and H. Ney (1994), "Leaving *m* samples out: Generalizing the Turing–Good formula", In preparation.

H. Ney (1990), "Acoustic modelling of phoneme units for continuous speech recognition", *Proc. EUSIPCO-90 Fifth European Signal Processing Conf., Barcelona, September 1990,* pp. 65–72.

H. Ney (1993a), "Search strategies for large-vocabulary continuous-speech recognition", *Proc. NATO Advanced Study Institute on Speech Recognition and Understanding, Bubion, Spain, 1993,* In print.

H. Ney (1993b), "Modeling and search in continuous-speech recognition", *Proc. European Conf. Speech Communication and Technology, Berlin, September 1993,* pp. 491–500.

H. Ney and U. Essen (1991), "On smoothing techniques for bigram-based natural language modelling", *Proc. Internat. Conf. Acoust. Speech Signal Process., Toronto, May 1991,* pp. 825–828.

H. Ney, R. Haeb-Umbach, B.-H. Tran and M. Oerder (1992a), "Improvements in beam search for 10000-word continuous speech recognition", *Proc. Internat. Conf. Acoust. Speech Signal Process., San Francisco, CA, March 1992,* pp. I-9–12.

H. Ney, D. Mergel, A. Noll and A. Paeseler (1992b), "Data driven organization of the dynamic programming beam search for continuous speech recognition", *IEEE Trans. Signal Processing,* Vol. SP-40, No. 2, February, pp. 272–281.

H. Ney, U. Essen and R. Kneser (1994a), "On structuring probabalistic dependencies in stochastic language modelling", *Computer Speech Language,* Vol. 8, pp. 1–38.

H. Ney, V. Steinbiss, R. Haeb-Umbach, B.-H. Tran and U. Essen (1994b), "An overview of the Philips research system for large-vocabulary continuous-speech recognition", *Internat. J. Pattern Recognition and Artificial Intelligence,* Vol. 8, No. 1, pp. 33–70.

M. Oerder and H. Aust (1994), "A realtime prototype of an automatic inquiry system", *Proc. ICSLP'94 Internat. Conf. Spoken Language Processing, Yokohama, 1994,* pp. 703–706.

M. Oerder and H. Ney (1993), "Word graphs: An efficient interface between continuous-speech recognition and language understanding", *Proc. Internat. Conf. Acoust. Speech Signal Process.'93, Minneapolis, MN,* pp 119–122.

D. Paul and J. Baker (1992), "The design for the Wall Street Journal-based CSR corpus", in *DARPA Speech and Language Workshop* (Morgan Kaufmann, San Mateo, CA).

H.-W. Ruehl, S. Dobler, J. Weith, P. Meyer, A. Noll, H.H. Hamer and H. Piotrowski (1991), "Speech recognition in the noisy car environment", *Speech Communication,* Vol. 10, No. 1, February, pp. 11–22.

Schwartz, R. and S. Austin (1991), "A comparison of several approximate algorithms for finding multiple (*N*-BEST) sentence hypotheses", *Proc. Internat. Conf. Acoust. Speech Signal Process.'91, Toronto, Canada,* pp. 701–704.

V. Steinbiss, B.-H. Tran and H. Ney (1994), "Improvements in beam search", *Proc. ICSLP Internat. Conf. Spoken Language Processing 1994, Yokohama, Japan,* pp. 2143–2146.