High-Level Approaches to Confidence Estimation in Speech Recognition

Stephen Cox, *Member, IEEE,* and Srinandan Dasmahapatra

***Abstract*—We describe some high-level approaches to estimating confidence scores for the words output by a speech recognizer. By "high-level" we mean that the proposed measures do not rely on decoder specific "side information" and so should find more general applicability than measures that have been developed for specific recognizers. Our main approach is to attempt to decouple the language modeling and acoustic modeling in the recognizer in order to generate independent information from these two sources that can then be used for estimation of confidence. We isolate these two information sources by using a phone recognizer working in parallel with the word recognizer. A set of techniques for estimating confidence measures using the phone recognizer output in conjunction with the word recognizer output is described. The most effective of these techniques is based on the construction of "metamodels," which generate alternative word hypotheses for an utterance. An alternative approach requires no other recognizers or extra information for confidence estimation and is based on the notion that a word that is semantically "distant" from the other decoded words in the utterance is likely to be incorrect. We describe a method for constructing "semantic similarities" between words and hence estimating a confidence. Results using the U.K. version of the Wall Street Journal are given for each technique.**

***Index Terms*—Confidence measures, latent semantic analysis, N-best lists, phoneme recognition, speech recognition.**

I. Introduction

THERE has recently been considerable research activity in the field of confidence estimation for the output of speech recognizers, for instance [1]–[4]. There are several motivations for attaching a measure of confidence to the words output by the recognizer: it can be used to improve the efficiency of a speech dialogue/understanding system by requesting confirmation or re-input of uncertain words, for detection of out-of vocabulary (OOV) words, to aid unsupervised speaker adaptation etc.

Many approaches to deriving confidence measures (CMs) for words have been based on using "side-information" derived from the decoder, such as normalized likelihoods [3], different decodings [2], number of competitors at the end of a word [4], etc. This information is often used as a feature vector for a classifier that attaches a probability to each output word that it is correct, e.g., [4]. We think that an approach that relies less on the details of the recognizer might be useful, for two reasons: first, speech APIs are usually now effectively "black boxes" and the side information necessary to construct many of the confidence measures described in the literature is not available; and secondly, it has been our experience that the performance of some of these CMs varies from recognizer to recognizer depending on details of the design of the decoder. Our motivation in developing the techniques described here is to make the estimation of confidence measures less recognizer-specific by using extra information in which the acoustic and language models are less tightly coupled than they are in the recognizer. If side-information *is* available from the recognizer, it can be combined with the information generated using these approaches to generate more powerful CMs.

An important objective in this work is to attempt to isolate the language and acoustic modeling components of the recognizer in order to assess separately the evidence for decoding a particular segment of the speech as a sequence of words. This is motivated, among other observations, by the following result of an experiment. We observed that the conditional probability of a word being correct or incorrect depends on the correctness or incorrectness of the word preceding it (the same phenomenon was observed in [5]). Moreover, this deviation from statistical independence increased monotonically as the language model scaling factor was made bigger. This suggested to us that a useful way of tackling the problem of confidence estimation was to disentangle the correlative effects of the language model. A complete de-coupling of the acoustic and language modeling components is not possible. However, in this paper, we have attempted to weaken the word-internal phonotactic constraints imposed by the word recognizer which can override acoustic cues that might otherwise indicate a different phone-level decoding. We then try to correlate the result of such a recognition with that of a word-level recognizer with the degree of correlation taken to suggest degrees of confidence.

This approach points the way toward a system-independent method for computing a confidence score. We isolate the acoustic modeling information by using a phone recognizer working in parallel with the word recognizer, an approach also used in [6],[7]. Hypotheses from the phone recognizer are not constrained by the language model and so provide a decoding that is based purely on the likelihood of the speech data having been produced by the acoustic models. The use of an additional decoder is not impractical, as an appropriately configured phone recognizer imposes a very small computational load compared with a large vocabulary word recognizer. The phone recognizer hypotheses may then be combined in several ways

Manuscript received July 27, 2001; revised June 24, 2002. This work was funded by the U.K. Engineering and Physical Sciences Research Council under Grant GR/L81253. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Shrikanth Narayanan.

S. Cox is with the School of Information Systems, University of East Anglia, Norwich, U.K.

S. Dasmahapatra was with the School of Information Systems, University of East Anglia, Norwich, U.K. He is now with the Department of Electronics and Computer Science, University of Southampton, Southampton, U.K.

Digital Object Identifier 10.1109/TSA.2002.804304

(e.g., with hypotheses from the word recognizer, with the language model) to assess confidence in decodings from the word recognizer.

We have also examined a technique that takes the high-level approach to its limit in that it requires only the top hypothesis output by the word recognizer to generate a confidence measure for each decoded word with no ancillary information from the speech recognizer. The idea is that a word that is semantically unrelated to the other decoded words in the utterance is likely to be incorrect. An N-gram language model incorporates only "short range" semantic coherence, inasmuch as words that have a high probability of occurring together are sometimes semantically linked. Hence, a measure of the "semantic similarity" of every decoded word to the other decoded words is independent information that can be included in the decision process. We describe a technique for estimating this "semantic similarity" and show how it can be used to provide a confidence measure.

The paper is structured as follows. Section II describes the data and the recognizers used throughout the experiments documented in the paper. In Section III, we introduce the notation used and develop the mathematical ideas underlying the "phone correlation" techniques. These techniques are described in Section IV and results are given for them. Section V gives the motivation and the theory behind the development of "metamodels" and also gives results for the technique. Section VI discusses the semantically inspired approach to confidence estimation, describes the techniques developed and gives results. In the final section, Section VII, we summarize the ideas and main results in the paper and discuss future work in this area.

II. Speech Data and Recognizers Used in These Experiments

The baseline recognizer was trained on speech data from the U.K. version of the *Wall Street Journal* database, WSJCAM0 data-set [8], using mainly "standard" techniques implemented in the Entropic HTK package. The data was parameterized to a 39-D vector consisting of 12 MFCCs and a log energy value + velocity + acceleration coefficients. The specifications of the recognizer were as follows:

1) trained on the speaker-independent training set `si_tr` of WSJCAM0 (92 talkers, $\sim$ 90 utterances per speaker);
2) number of words in vocabulary $\sim$ 20 000;
3) Bigram language model (trained on 60M words from the North American business news corpus), perplexity $\sim$ 160;
4) 3500 states created by tree-clustering word-internal triphones; eight Gaussian mixture components per state;
5) three-state left-to-right models;
6) test set used: the speaker-independent development set `si_dt` in WSJCAM0, $\sim$ 1800 utterances;
7) performance (word level): 74.0% correct, 68.2% accurate.

The phone recognizer was also trained on the complete WSJCAM0 training-set and consisted of 45 monophone models, three states to a model with no skips. As for the word recognizer, each HMM state had a Gaussian mixture model of eight components associated with it. Performance was 70.4% correct and 49.5% accurate.

The development set of the database was divided into a set of 913 sentences for training our confidence estimators and another 913 for testing. Each word in the recognition output from each sentence was tagged as "C" (correct) or "I" (incorrect) before being used in the CM experiments (see Section IV-C for more details). The HMM training/decoding software used for all experiments was HTK v2.2. [9].

III. Theoretical Background

The use of an unconstrained phone recognizer which works in parallel with the word recognizer allows a more general approach to confidence estimation. By "unconstrained", we mean that the phoneme decoder is configured using a simple network that allows any phoneme to follow any other with equal probability (a "phone-loop"). The question is then how to make the best use for confidence estimation of the extra information from the phone recognizer, which is independent of the language model. In this section, we develop some theory that enables us to propose two confidence measures that make use of this extra information: phone-correlation [10] and metamodels [11]. These techniques are described in detail in Sections IV and V.

A. Notation

The following notation is used in this paper:

1) $\mathbf{A}$ is the acoustic signal input into the recognizer;
2) $\mathcal{P} = \{p_1, p_2, \ldots\}$ is the set of phonemes;
3) $\boldsymbol{\mathcal{P}} = \{\mathbf{p}_1, \mathbf{p}_2, \ldots\}$ is the set of strings of phonemes;
4) $\mathcal{W} = \{w_1, w_2, \ldots\}$ is the vocabulary, i.e., the set of lexical entries;
5) $\boldsymbol{\mathcal{W}} = \{\mathbf{w}_1, \mathbf{w}_2, \ldots\}$ is the set of strings of lexical entries (i.e., strings of words);
6) $\mathbf{w}^* \in \boldsymbol{\mathcal{W}}$ is the output of the word recognizer in response to an utterance;
7) $\mathbf{q}^* \in \boldsymbol{\mathcal{P}}$ is the transcription of $\mathbf{w}^*$ into a sequence of phones;
8) $\mathbf{p}^* \in \boldsymbol{\mathcal{P}}$ is the output of the phone-loop recognizer in response to an utterance.

B. Decoupling the Acoustic and Language Models

For speech recognition, we attempt to find the word sequence $\mathbf{w}^*$ for which the probability $\Pr(\mathbf{w}|\mathbf{A})$ is largest among all word-sequences $\mathbf{w} \in \boldsymbol{\mathcal{W}}$. The decoding is performed in the standard way by expanding a word network into triphone sequences by dictionary lookup. Since the triphone sequences form a proper subset of $\boldsymbol{\mathcal{P}}$, we may invoke completeness and rewrite $\Pr(\mathbf{w}|\mathbf{A})$ as

$$\Pr(\mathbf{w}|\mathbf{A}) = \sum_{\mathbf{p} \in \boldsymbol{\mathcal{P}}} \Pr(\mathbf{w}|\mathbf{p}) \Pr(\mathbf{p}|\mathbf{A}). \tag{1}$$

The second term $\Pr(\mathbf{p}|\mathbf{A})$ is evaluated in a phoneme classification task, while the first term can be estimated from a pronunciation model $\Pr(\mathbf{p}|\mathbf{w})$ using Bayes theorem and a word segmentation algorithm.

In Viterbi decoding, the beam threshold placed on log-likelihoods restricts the sum above to an even smaller subset of $\mathcal{P}$

$$\Pr(\mathbf{w}|A) \approx \sum_{\mathbf{p} \text{ within beam}} \Pr(\mathbf{w}|\mathbf{p}) \Pr(\mathbf{p}|A). \quad (2)$$

The word recognition problem is to find $\mathbf{w}^*$ which maximizes (2) and whose phoneme sequence is $\mathbf{q}^*$.

Instead of choosing this subset of $\mathcal{P}$, we may instead choose the sequence $\mathbf{p}^*$ obtained from a phoneme classification task

$$\begin{aligned} \mathbf{p}^* &= \arg\max_{\mathbf{p}\in\mathcal{P}} \Pr(\mathbf{p}|A) \\ &= \arg\max_{\mathbf{p}\in\mathcal{P}} \Pr(\mathbf{A}|\mathbf{p}) \Pr(\mathbf{p}). \end{aligned} \quad (3)$$

In this paper, we set all phonemes to be equiprobable and statistically independent (a phone loop), so the Viterbi search in this problem ranges over sequences not considered in the word recognition problem as it disregards both intraword (phonotactic) and interword (language model) constraints.[1] Thus, a different approximation of (1) gives

$$\Pr(\mathbf{w}|A) \approx \Pr(\mathbf{w}|\mathbf{p}^*) \Pr(\mathbf{p}^*|A). \quad (4)$$

The knowledge of $\mathbf{p}^*$ from a phone-recognition experiment may be used to obtain an *alternative decoding* of the utterance by finding the word sequence $\mathbf{w}'$ that maximizes each side of (4)

$$\begin{aligned} \mathbf{w}' &= \arg\max_{\mathbf{w}\in\mathcal{W}} \Pr(\mathbf{w}|\mathbf{p}^*) \Pr(\mathbf{p}^*|A) \\ &= \arg\max_{\mathbf{w}\in\mathcal{W}} \Pr(\mathbf{w}|\mathbf{p}^*). \end{aligned} \quad (5)$$

Now that the lexical constraints on the choice of phonemes have been lifted, we shall try to gauge the correctness of the word-string $\mathbf{w}^*$ by comparing it with the phone-string $\mathbf{p}^*$. We can use (5) to make correlative comparisons at the word level, which we shall do in Section V. In the following section, we shall propose a confidence measure by correlating phone-strings alone.

To do that, we need to confront the problem of segmenting phone strings into words. We make the simplifying assumption that in order to assess the correctness of a word given the phone evidence, it is sufficient to consider the segments of $\mathbf{q}^*$ and $\mathbf{p}^*$ that correspond to a recognized word. If the decoded word sequence consists of N words, i.e., $\mathbf{w}^* = w_1, w_2, \ldots w_N$ then $\mathbf{q}^* = \mathbf{q}_1, \mathbf{q}_2, \ldots \mathbf{q}_N$ and $\mathbf{p}^* = \mathbf{p}_1, \mathbf{p}_2, \ldots \mathbf{p}_N$, where $\mathbf{q}_k$ is the dictionary transcription of the kth decoded word and $\mathbf{p}_k$ is the sequence of phonemes decoded by the phone decoder that corresponds to $\mathbf{q}_k$. We give the details of this correspondence in the following section.

[1]An imposition of a suitably constraining model of phone-sequence probabilities could be an alternative strategy, which would only leave the interword probabilities used in the word-recognition problem unconstrained. This would also increase the cardinality of the set obtained by intersecting the search spaces of the word-recognition problem and that of the phonologically constrained setting.

IV. Phone Correlation Techniques

The "phone correlation" techniques described in this section are based on using an estimate of $\Pr(\mathbf{q}_k|\mathbf{p}_k)$ as a measure of confidence for decoded word w_k. The recognizer's dictionary is used to rewrite w_k as a phone sequence $\mathbf{q}_k$—if there are multiple pronunciations of any word decoded in $\mathbf{w}^*$, it is necessary to find which one was decoded and use the phoneme string corresponding to that pronunciation.

A. Alignment and Comparison Methods

To estimate $\Pr(\mathbf{q}_k|\mathbf{p}_k)$, the phone streams $\mathbf{q}^*$ and $\mathbf{p}^*$ need to be correctly aligned to each other. Two methods for alignment were tested.

Frame-Level, FL. $\mathbf{q}^*$ was used together with the phone boundaries available from the word recognizer to tag each frame of an utterance with the identity of the phone decoded at that time. The frames were similarly tagged using the phone recognizer boundaries and the phones in $\mathbf{p}^*$. Hence, the tags applied by both word and phone recognizers to a sequence of frames corresponding to a decoded word could be compared on a frame by frame basis.

Phone-Level, PL. Dynamic programming was used to align $\mathbf{q}^*$ with $\mathbf{p}^*$. The sequence of phones $\mathbf{q}_k$ corresponding to word w_k decoded by the word recognizer could then be aligned with $\mathbf{p}_k$. When this method is used, it is possible for some phones decoded by one recognizer not to be paired by the dynamic programming algorithm to a phone decoded by the other recognizer. These phones are regarded as "insertions" and are ignored in subsequent analysis.

Two methods of comparing the tags on the frame-aligned sequences (FL) or the DP-aligned sequences (PL) were used. For word w_k, let the corresponding subsequences be $\mathbf{q}_k = q_1 \ldots q_{Q_k}$ and $\mathbf{p}_k = p_1 \ldots p_{P_k}$. For the frame-aligned method, the length of the subsequences $P_k = Q_k$, as they measure the duration of the phoneme in the decoded speech stream; for the method in which the phonemes are aligned by dynamic programming, the same condition entails once the insertions are ignored, except Q_k now refers to the number of phonemes in word w_k in the lexicon.

1) Cross confusion matrix measure, CCM. Assuming that the phoneme occurrences are independent, we define a confidence measure for word w_k as

$$CCM(w_k) = \sum_{l=1}^{Q_k} \log \Pr(q_l|p_l). \quad (6)$$

The term $\Pr(q_l|p_l)$ is estimated by forming a "cross confusion matrix" $\Psi(q,p)$ (for $p, q \in \mathcal{P}$) between the phone-loop and the word recognizers as follows:

a) for each utterance, use dynamic programming to align $\mathbf{q}^*$ with $\mathbf{p}^*$;
b) to make entry $\Psi(q,p)$, count the number of times that phone $q_l = q$ was aligned with phone $p_l = p$;
c) use $\Psi(q,p)$ to estimate $\Pr(q|p)$.

The CCM measure is based on the premise that the more the phone-loop and the word recognizer agree, the more chance

there is of a correct decoding of the word. A simple way of quantifying this would be define a confidence C_l for the lth phone in a word as

$$C_l = \begin{cases} 1, & \text{if } q_l = p_l \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

where q_l and p_l are the identities of the lth pair of aligned phonemes. This was tested, but found to be a poor measure, even when zero in the (7) was replaced by an empirically determined floor value. An alternative premise is that a *consistent* alignment between pairs of phonemes may indicate correct decoding of a word, even if the two phones aligned are not the same. A consistent co-occurrence of a pair of phonemes is expressed by a higher than average value of $\Pr(q_l|p_l)$, which increases the value of $CCM(w_k)$ in (6). Hence, $CCM(w_k)$ should be higher for correctly decoded than for incorrectly decoded words.

2) Likelihood-ratio, LR. The CCM measure does not take into account whether the phones decoded by either recognizer are correct or not. We extend CCM by using the training data to form *two* cross confusion matrices, Ψ_C made using phone co-occurrences associated with correctly decoded words and Ψ_I made using phone transcriptions associated with incorrectly decoded words. (Note that when w_k is correctly decoded, $\mathbf{q}_k$ is the reference transcription for w_k; hence, Ψ_C is effectively an estimate of the confusion matrix of the phone recognizer.) Using Ψ_C and Ψ_I, we estimate $\Pr(q|p, w = C), w \in \mathcal{W}$ (where q is in word w and p is the corresponding phoneme from the phone recognizer) and $\Pr(q|p, w = I)$. Using Bayes theorem, we find

$$\Pr(w = C|q,p) = \frac{\Pr(q|p, w = C)\Pr(w = C)}{\Pr(q|p)}$$

$$\text{and} \quad \Pr(w = I|q,p) = \frac{\Pr(q|p, w = I)\Pr(w = I)}{\Pr(q|p)}.$$

Hence, a correct/incorrect likelihood ratio L_l^k for the co-occurrence of a pair of phones q_l and p_l corresponding to the word w_k can be defined

$$\begin{aligned} L_l(w_k) &= \frac{\Pr(w_k = C|q_l, p_l)}{\Pr(w_k = I|q_l, p_l)} \\ &= \frac{\Pr(q_l|p_l, w_k = C)}{\Pr(q_l|p_l, w_k = I)} \frac{\Pr(w_k = C)}{\Pr(w_k = I)}. \end{aligned} \tag{8}$$

(Note that a similar approach of using a likelihood ratio based on a cross-confusion matrix was taken in [12].) Again assuming independence of terms, a confidence measure for word w_k is then

$$\begin{aligned} LR(w_k) &= \sum_{l=1}^{Q_k} \log L_l(w_k) \\ &= Q_k\theta + \sum_{l=1}^{Q_k} \log \frac{\Pr(q_l|p_l, w_k = C)}{\Pr(q_l|p_l, w_k = I)} \end{aligned} \tag{9}$$

where θ is estimated as the logarithm of the ratio of correct words to incorrect words.

Both these methods can be equally well applied to the sequences of tagged frames rather than aligned phones.

Using the training-data, CCM and LR measures were computed for each training-set word and histograms of the values for "C" and for "I" utterances were estimated. The threshold at which the C/I tagging accuracy was maximized (using a Bayesian approach) on the training-set was then determined.

B. Baseline Confidence Measures: The "N-Best" Technique

The techniques described in Section IV-A were benchmarked against a "standard" confidence measure estimating technique, the N-best method [2]. This technique effectively uses an estimate of the stability of each decoded word in the word lattice, the premise being that words that are stable in the lattice are more likely to be correct than those that occur infrequently and/or erratically. An "N-best" confidence measure was computed for each word in the top hypothesis from the word recognizer in the following way:

1) dynamic programming is used to align the top hypothesis $\mathbf{w}^*$ to each of the next 99 hypotheses from the word recognizer;
2) the number of times M_k that word w_k in the top hypothesis occurs in the same alignment position in the other hypotheses is counted;
3) the confidence measure for word w_k is estimated as $M_k/100$;
4) the optimum threshold for C/I tagging accuracy is set as stated in Section IV-A.

It could be argued that the N-best technique does not perform as well as some others in the literature. However, N-best lists are available from many recognizers and the technique for processing them into confidence scores is fairly standard. As a consequence, N-best is the closest technique we currently have to a recognizer-independent and standard baseline for deriving CMs and hence is an appropriate baseline for benchmarking the performance of the recognizer-independent CMs described here.

C. Results for the Phone-Correlation Techniques

Various methods for rating confidence measures have been proposed [13]. The measure adopted here is the confidence error rate (CER) [14], which is just the error-rate on the task of tagging each decoded word as either "C" or "I." Clearly, if the word error rate of the recognizer is e, a CER of e can be obtained by guessing each word as "C." Hence, we use the percentage improvement in CER over guessing provided by the confidence measure which we define as the *confidence gain* (CG), i.e.,

$$CG = \frac{CER_G - CER_{CM}}{CER_G} \times 100\% \tag{10}$$

where CER_G is the "guessing" CER and CER_{CM} the CER obtained using the confidence measure.

The baseline performance of our word recognizer is 74.0% correct, 68.2% accurate. To measure the CER when no CM is used, it is required to tag every output word as either "C" or "I." Hence, substitutions and insertions in the word recognizer output are marked as "I" and deletions are ignored. The adjusted word error rate is 31.0% and this corresponds to CER_G.

TABLE I
PERFORMANCE OF PHONE CORRELATION TECHNIQUES

Technique	CER_{CM} (%)	CG (%)
FL + CCM	30.2	2.6
FL + LR	30.3	2.3
PL + CCM	29.8	3.9
PL + LR	29.2	5.8
N-best	24.1	22.2

Table I shows the performance of the techniques. Using a DP phone alignment (PL) of the two phone streams is superior to using a frame-level alignment and this is most effective when used in conjunction with the likelihood ratio (LR). Although the results from the techniques using a parallel phone recognizer are statistically significant, they are not practically significant and none comes close to the performance of the baseline N-best technique. It was conjectured that if the acoustic models used in the two recognizers had some independence, these correlation techniques might perform better. We have some evidence that this is the case, but have not been able to confirm our results in time for publication.

V. METAMODELS

An attractive alternative to correlating the phone hypotheses from the recognizers is to construct word hypotheses from the phone recognizer output and compare these hypotheses with those output by the word recognizer. The alternative decoding is obtained from (5) by some segmentation algorithm which incorporates the classification errors inherent in $\mathbf{p}^*$. Again, we would expect that where both sets of hypotheses decoded the same word in the same position, confidence in the correctness of that word would be high, given that the phoneme strings in the word decoding problem and in the phone recognition problem are from different "slices" of $\boldsymbol{\mathcal{P}}$ imposed by the respective Viterbi approximations. We first attempted to segment $\mathbf{p}^*$ into words using a sliding window to isolate fixed-length subsequences of $\mathbf{p}^*$ that might constitute fragments of a word, together with a confusion matrix that enabled us to account for possible errors in the phone decoding (a similar approach was used in [15]). However, these experiments convinced us that the large number of errors, especially insertions, in the phone recognizer stream made the segmentation problems quite unwieldy [10].

We therefore sought a method of accounting for likely errors in the phone recognizer stream whilst simultaneously segmenting it to form word hypotheses. A word recognition system effectively uses the Shannon "noisy channel" paradigm and this approach seemed to be also appropriate for our task. In a word recognition system, the HMMs model the mapping from sequences of "noisy" feature vectors to phonemes and the language model ensures that only legal sequences of phonemes (i.e., sequences of words) can be decoded. We use the same principle to decode the stream from the phone recognizer into words. However, the input to our word recognizer is the phone stream $\mathbf{p}^*$ rather than a stream of acoustic feature vectors and our HMMs model the mapping between the "noisy" phone stream output by the phone recognizer and the reference transcription. As these discrete HMMs model the output from acoustic HMMs, we have termed them *metamodels* and the associated decoder a *meta-recognizer*.

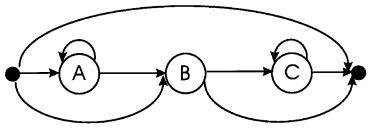

Fig. 1. Architecture of the metamodel of a phoneme.

Fig. 1 shows the architecture of a metamodel of a phoneme. The central state *B* of the metamodel for a phoneme models correct decodings and substitutions of this phoneme made by the phone recognizer. The outer states, *A* and *C*, model (possibly multiple) insertions. A deletion is accommodated by the loop from input to output. Each state has a discrete probability distribution over the phonemes associated with it. The metamodels are trained in the same manner as acoustic HMMs using embedded Baum-Welch re-estimation as follows:

1) Set the nonzero model transition probabilities to random values and seed the symbol distributions for each state equiprobably;
2) Use the reference transcriptions to concatenate the appropriate sequence of phoneme metamodels for each training set utterance;
3) Use the decoded string $\mathbf{p}^*$ for each utterance as training data and iterate re-estimation of the transition probabilities and distribution probabilities of the metamodels, using the Baum-Welch algorithm, until a suitable convergence criterion is reached.

Since the phone recognizer tends to produce insertions, there are always enough phones in any utterance to train the sequence of metamodels.

The language model is then used to compile the meta-recognizer network, which is identical to the network used in the word recognizer except that the nodes of the network are the appropriate metamodels rather than the triphone acoustic models used by the word recognizer. At recognition time, the output of the phone recognizer $\mathbf{p}^*$ is passed to the meta-recognizer to produce a set of hypotheses.[2] Dynamic programming is used to compare these hypotheses with the top hypothesis from the word recognizer and any word in the word recognizer output that occurs more than N times in the same position in the meta-recognizer output is marked as "C." N is determined experimentally and is in the range 1–5.

A. Metamodel Results

Table II compares the results for the metamodels confidence score with those obtained using N-best. Note that a slightly smaller test-set from that used in Section IV-C was used here and the result for N-best is slightly better than that quoted in

[2]When formulating word hypotheses using the meta-recognizer, the acoustic/language modeling balancing factor α is set to the same value as is used to give optimal recognition results from the word recognizer. This may not be optimal for the metarecognizer, nor is it guaranteed to be optimal for the confidence classification problem.

TABLE II
PERFORMANCE OF METAMODELS CM COMPARED WITH N-BEST CM

Technique	CER_{CM} (%)	CG (%)
Nbest	23.9	24.8
Metamodels	21.9	31.1

TABLE III
ERROR PATTERN FOR METAMODELS AND N-BEST CMS

	Metamodel C	Metamodel I
N-best C	4413	1120
N-best I	1225	463

TABLE IV
PERCENTAGE OF CORRECT AND INCORRECT WORDS (COLUMNS 3 AND 4) COMPARED WITH PREDICTION OF TWO CONFIDENCE MEASURES

N-best tag	metamodel tag	prob C%	prob I%
C	C	91.8	8.2
C	I	50.3	49.7
I	C	59.2	40.8
I	I	14.5	85.5

Section IV-C. Table III lists the number of words tagged correctly or incorrectly for each of the two confidence measures for the subset of words for which both measures gave a tag (some utterance files had to be discarded because the pruning thresholds for recognition were set too tightly). Table III shows, for example, that 1225 words were mistagged by N-best, but correctly tagged by the metamodel confidence measure etc. It is promising that only 463 of the 7221 words listed above were mistagged by both measures, indicating an upper bound of 6.4% tagging error over the baseline guessing measure (31% error) possible by some combination of the two features. A further breakdown of these figures in order to compare the performance of each tag-pair is given in Table IV. Table IV shows that when both confidence measures tag a word as correct, there is a 91.8% chance that the decoded word is correct. Conversely, when both tag incorrect, there is a 85.5% chance that the word is incorrect.

Receiver operating curves for the N-best and for the metamodel confidence methods are shown in Fig. 2.

B. Discussion

In Fig. 2, the lack of points for metamodel CMs at low false alarm levels is due to the fact that a metamodel CM cannot be computed for all words decoded by the word recognizer, because not all the words in the word recognizer output string appear in the word strings hypothesised by the metarecognizer. However, Fig. 2 shows that metamodel CMs are superior to N-best CMs in regions where they operate together. In fact, the N-best and metamodel methods are complementary, because it is not possible to operate the N-best method at low levels of false acceptance, as there are some incorrect words that appear in every hypothesis and hence have a confidence of 1.0 (a problem also noted in [16]). Hence, metamodels and N-best techniques can be used in combination to enable operation over the full range of false alarm/false acceptance tradeoff.

VI. USING SEMANTIC INFORMATION AS A WAY OF MEASURING CONFIDENCE

An approach to confidence estimation that requires neither side information from the recognizer nor a parallel phonetic recognizer must rely on either syntactical or semantical clues. Operating under the premise that correct recognition results should be more grammatical than incorrect results, Zhang and Rudnicky have used a parser to extract confidence measures [17]. In this section, we describe the use of "semantic" knowledge to decide whether a decoded word is likely to be correct or not. Palmer and Ostendorf made some use of semantic knowledge in [5], where they found that knowing whether a word was part of a "location, organization or person phrase" was useful. Pao *et al.* also used semantic information [18], but their technique required hand-coding of the semantic categories of the vocabulary words, which ours does not.

Our technique was inspired by the observation that humans can identify some words that are incorrect in recognizer output on semantic grounds. Consider, for instance, a sentence decoded by our recognizer: "Exxon corporations said earlier this week that it replaced one hundred forty percent its violin gas production in nineteen eighty serve on." Here, the word "violin" is clearly incorrect because it is not semantically related to the other content words in the sentence (the correct transcription is "oil and"). Of course, not all incorrectly decoded words can be so clearly identified on semantic grounds as in this example, but the occurrence of "semantic outliers" is not infrequent in recognizer decodings (see Section VI-A for some numbers from our own recognizer). If decoded words could be tagged using this principle, no side information from the recognizer would be required. If side information *is* available, it can be used in conjunction with the semantic information, as this information will be independent of any information about the quality of the acoustic matching in the decoder and will also have a degree of independence from any measures derived from the decoder's language model. In Section VI-F, we describe how these two sorts of information were combined to give an improved CM.

A. Preliminary Experiment

In a preliminary experiment, we investigated the viability of the idea of estimating confidence in the correctness of a decoded word using semantic information. We examined about 600 sentences decoded by our recognizer from the WSJCAM0 corpus and, without knowing the correct transcription, attempted to mark the words that we thought were incorrect on "semantic" grounds. This marking was done conservatively, i.e., only words that seemed to be clearly wrong because they were incongruous were tagged as I. The confusion-matrix of our hand-marking is shown in Table V. Table V indicates that $421+49=470$ words were tagged as I out of the $2720+421=3141$ that were actually incorrect, i.e., $Recall = 15\%$. Of the 470 words tagged as I, 421 were correctly tagged, so $Precision = 89.6\%$ (these definitions of *Recall* and *Precision* are used in Section VI-F). This experiment indicated to us that using a machine capable

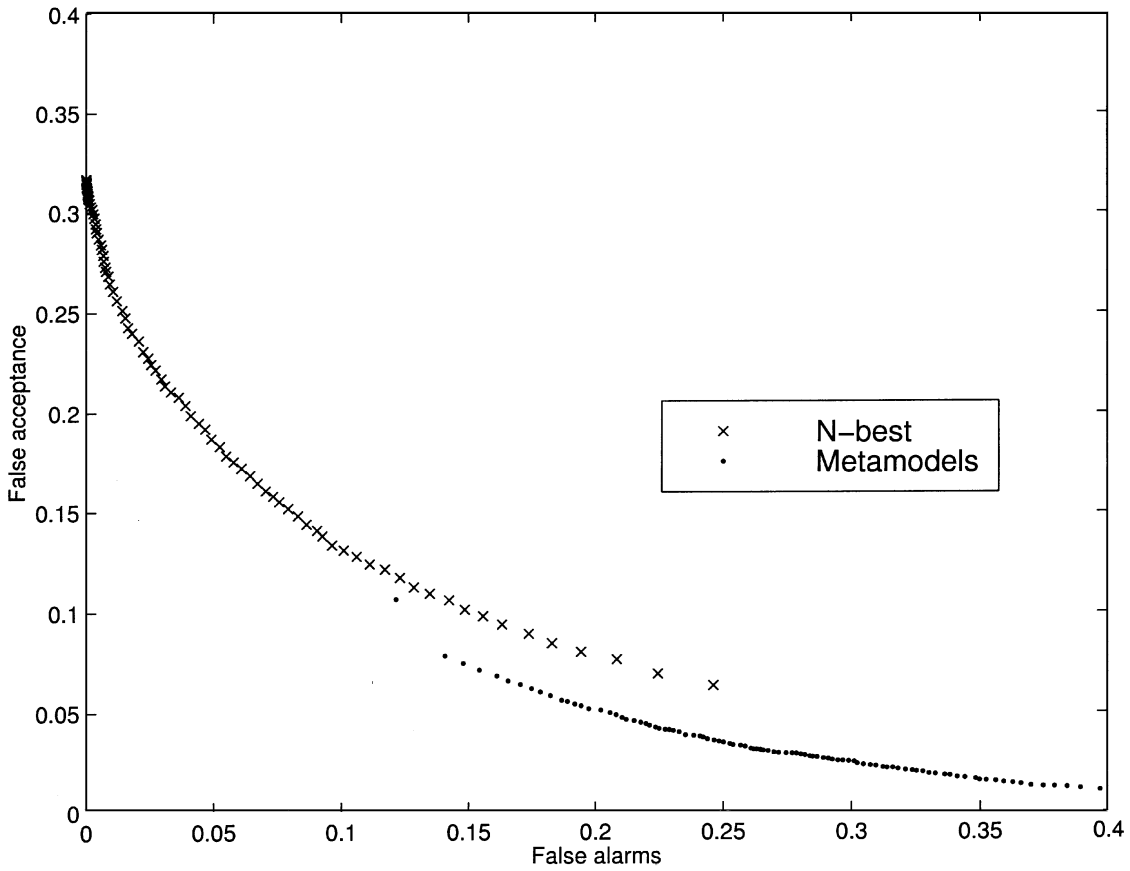

Fig. 2. False acceptances versus false alarms at different thresholds. The dots are for the metamodel operating points and the pluses (+) are for N-best.

TABLE V
CONFUSION-MATRIX FOR HAND-TAGGING OF WORDS ON SEMANTIC GROUNDS

		CLASSIFIED	
		Unclassified	Incorrect
ACTUAL	Correct	7680	49
	Incorrect	2720	421

of emulating human performance by using a "semantic" criterion to identify incorrect words had the potential to identify only a small number of words, but with fairly high precision. The words we identified were all uncommon nouns or verbs that were incongruous: it would not be possible to identify incorrectly decoded common words, such as function words. However, nonfunction words, in most cases, bear more information and so are generally more important if a confidence measure is needed to support, e.g., a dialogue system.

B. Application of Latent Semantic Analysis

Latent semantic analysis (LSA) is a technique that has been in use for some years in the field of information retrieval and has latterly been applied in speech recognition [19]. It is not proposed to describe the theory of LSA in detail here—for an introduction, see, for example, [20]. LSA is a technique for associating words that tend to co-occur within documents that are "semantically coherent" (examples of documents are entries in an encyclopaedia or stories in a newspaper). The assumption is that words that tend to co-occur across documents are semantically linked.

The essential idea behind LSA is to form a matrix of word/document co-occurrences and then to represent the row and column vectors of this very large matrix in a greatly reduced subspace using the technique of singular value decomposition (SVD). Because the matrix is very sparse and its rank is also much lower than its dimensionality, it is possible to represent the vectors in a low dimensional space with relatively small error. The key property of LSA is that words whose vectors are "close" in the reduced space correspond to semantically similar words (also, documents whose vectors are close in this space convey similar semantic meanings). It is this property that we have exploited for use in forming a confidence measure. By projecting the vocabulary words into the subspace, we can estimate a "semantic similarity" between any pair of words. These similarities can be used to provide an estimate of the likelihood of the words co-occurring within the same utterance, under the obvious assumption that the utterance and the training texts are homogeneous.

C. Application of LSA

The 1994 subset of the WSJ [available in the North American News Text Corpus (NANT)] was used to form a word/document matrix W for latent semantic analysis. A "document" was defined to be a news story and the text was preprocessed to remove punctuation and to spell out abbreviations, numbers, dates, etc. The entries in W were formed from the raw counts c_{ij} of the

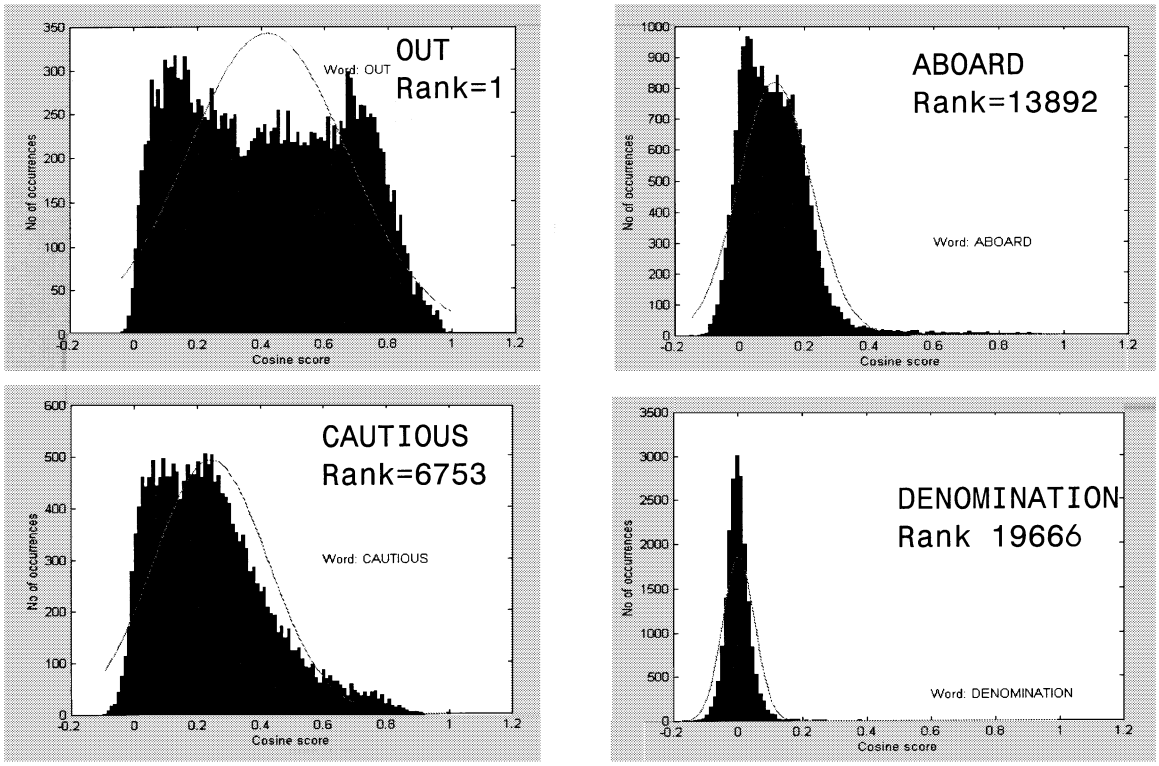

Fig. 3. Distributions of "semantic similarities" for four words.

number of times word i appears in document j according to the technique described in [19]. In [19], entry (i, j) of W is defined as $W_{ij} = G_i L_{ij}$, where G_i is a "global weight" for word w_i and reflects the fact that words occur with different frequencies through documents and L_{ij} a local term that adjusts c_{ij} to take account of widely different values. There were 19 685 different words and 19 396 documents in the corpus. SVD of W was done using the special MATLAB routine for sparse matrices `svds()`. After some experimentation with different dimensionalities of reduced space, a reduced space dimensionality of 100 was used. Each word was then described by a 100-D vector and the "semantic similarity" between words w_m and w_n in the lexicon, $S(m, n)$, was computed as the cosine of the angle between the vectors, i.e.,

$$S(w_m, w_n) = \frac{\boldsymbol{\omega}_m \cdot \boldsymbol{\omega}_n}{|\boldsymbol{\omega}_m|\,|\boldsymbol{\omega}_n|} \tag{11}$$

where $\boldsymbol{\omega}_m$ is the 100-D vector associated with word w_m. It should be noted that $-1 \leq S(m, n) \leq 1$ and a high positive value for $S(m, n)$ indicates that the words w_m and w_n have a high semantic correlation.

Before proceeding to use these similarities for estimation of confidence measures, we were interested to examine their distributions for different words. The similarity between each word in the lexicon and all other words was computed to give a set of similarities $\{L_i\}$ for word w_i. The mean similarity $\bar{L}_i$ was computed for each word and the words were then ranked by their value of $\bar{L}_i$. It was noticeable that the words with high values of $\bar{L}_i$ were predominantly (but not exclusively) function words. The words with the lowest values were, without exception, rarely occurring nouns. Fig. 3 shows the distribution of the $\{L_i\}$ for four words at different positions in the ranking. The implication of Fig. 3 is that broad distributions are associated with words that occur with many other words and hence have a broad range of semantic similarities, whereas the narrower distributions centred on zero represent words that occur with only a small fixed set of words and hence have a semantic similarity close to zero to most other words in the lexicon.

It seemed possible that these "semantic similarities" were actually simply a reflection of the fact that a word that occurred often in the training data would naturally co-occur with many other words and hence have a high mean similarity, whereas the reverse would be true for infrequently occurring words. However, it was found that there was only a very weak correlation between the mean semantic similarity and the number of occurrences (or the ranking of the number of occurrences) of a word ($|r| = 0.15$ for the latter case). For example, the word "proving" has a medium number of occurrences in the training-data (91, rank 6992) but a high mean semantic similarity of 0.385 (rank 165), which indicates that it co-occurs with a diverse collection of words. Conversely, "gas" has a high number of occurrences in the training-data (1946, rank 579), but a very low mean semantic similarity of 0.02 (rank 19 081), which indicates that it co-occurs only with a very specific set of other words.

D. Confidence Measures From LSA

We first attempted to identify words whose meaning and usage were not cognate with the other decoded words in the sentence by using the precomputed semantic similarities to compute the mean semantic similarity for each decoded word. Suppose that the sequence of words decoded from an input utterance is $w_{U(1)}, w_{U(2)}, \ldots, w_{U(N)}$, where $U()$ maps from

the number of the decoded word in the sentence to the number of the same word in the lexicon. The mean semantic similarity for the ith decoded word is

$$MSS_i = \frac{1}{N}\sum_{j=1}^{N} S(U(i), U(j)). \tag{12}$$

(Notice that if word w_i is the same as word w_j, $S(U(i), U(j)) = 1$, which increases the value of MSS_i. In practice, only common function words usually reoccur in a decoding and as these words will be discarded after the application of a stop list (see Section VI-E), this effect does not cause a problem). We would expect MSS to be low for semantically incorrect words and high for words that are cognate. However, MSS is a poor indicator of semantically incorrect words. The reason for its poor performance is that *most* words have high similarities to function words and although a semantically incorrect word may have low similarities to other content words in the decoded sentence, these low similarities are masked by the "noise" from the higher similarities to decoded function words.

E. Using a Stop List to Eliminate Common Words

This result suggested that it was necessary to discard decoded words that had high mean semantic similarities to most other words in the lexicon, as these words had low semantic weight and contributed mainly noise to the value of MSS for other words. Accordingly, we experimented with using a stop list [21] of decoded words whose value of $\bar{L}_i$ was above a threshold L_T. We then compute a confidence measure for each of the remaining words. The confidence measures we experimented with were as follows.

1) MSS as defined in (12);

2) MR, the mean rank of the semantic similarities to the decoded word w_i. MR_i was computed by finding the rank of each $S(U(i), U(j)$ in the set of semantic similarities $\{L_i\}$ and then computing the mean;

3) PSS, the probability that the set of semantic similarities between word w_i and the other decoded words was generated from the distribution of similarities $\{L_i\}$

$$PSS_i = \prod \Pr\left(L_i \le S(U(i), U(j))\right) \tag{13}$$

where L_i is a random variable whose distribution is estimated from $\{L_i\}$, the set of semantic similarities for word w_i. We approximated the distributions shown in Fig. 3 by five component Gaussian mixtures to estimate $\Pr(L_i \le S(U(i), U(j)))$.

In practice, we found that all three of the above statistics gave very similar performance, with PSS marginally the best. The effect of varying L_T on the tagging accuracy of PSS is shown in Table VI. In Table VI, CER_G is the error-rate of the recognizer on the retained words (= the "guessing" error-rate), CER_{PSS} is the tagging error-rate when using the PSS confidence-measure and CG is the confidence gain. It is interesting that CER_G at first decreases, probably as more function-like words, which tend to have a higher error-rate than nonfunction words, are discarded. However, when 78% of words are discarded, CER_G begins to rise again and continues to rise. Examination of the words retained when $L_T \le 0.15$ shows a greatly increased proportion of single letter words, mostly "u," "p," and "s." These words are almost always incorrect insertions by the decoder and hence the error-rate increases.

TABLE VI
EFFECT OF VARYING THE THRESHOLD L_T

Threshold L_T	% words discarded	CER_G	CER_{PSS}	CG (%)
0.45	0	0.303	0.288	5.0
0.4	5	0.296	0.282	4.7
0.35	25.7	0.286	0.274	4.2
0.3	45.9	0.260	0.247	5.0
0.25	53.6	0.238	0.221	7.1
0.2	64.6	0.230	0.222	3.4
0.15	78	0.243	0.242	0.4
0.1	88	0.276	0.285	-3.2
0.05	97	0.296	0.294	0.7

Table VI does not show the predicted increase in tagging accuracy when commonly co-occurring words are eliminated. An examination of the distribution of the values of PSS was revealing. In Fig. 4, the values of PSS for correctly (top) and incorrectly (bottom) decoded words are shown for $L_T = 0.25$ i.e., with 53.6% of decoded words excluded. The histograms have a large overlap showing that PSS is unable to separate these classes very effectively. However, it is interesting to examine the "tails" of the distributions. Our hypothesis is that a semantic confidence measure should be effective at identifying incorrect words and so we would expect to see a high probability of low values of PSS for *incorrect* words and a low probability of low values of PSS for *correct* words. In fact, Fig. 4 shows that the probabilities of low values of PSS are very similar for both correct and incorrect words. However, *high* values of PSS are significantly more probable for correct words than for incorrect words. Hence, PSS is deriving its discrimination by identifying *correctly* decoded words. Analysis revealed that the words associated with high values of PSS were predominantly words that commonly occurred in the WSJ data (numbers, financial terms, etc.) that were highly cognate with each other. Inspection of the decoded words that had very low values of PSS associated with them showed that some of these were commonly occurring words that had been correctly decoded. The most likely reason that these common words have low semantic similarity to other decoded words is that they have not been seen to co-occur in the training corpus with them.

F. Combining Semantic and N-Best Confidence Measures

The semantic confidence-measure based on the value of PSS is a weak indicator of correctly or incorrectly decoded words, but the information it provides is largely independent of similar information from the decoder itself. It seemed possible that combining PSS with a CM derived directly from the decoder

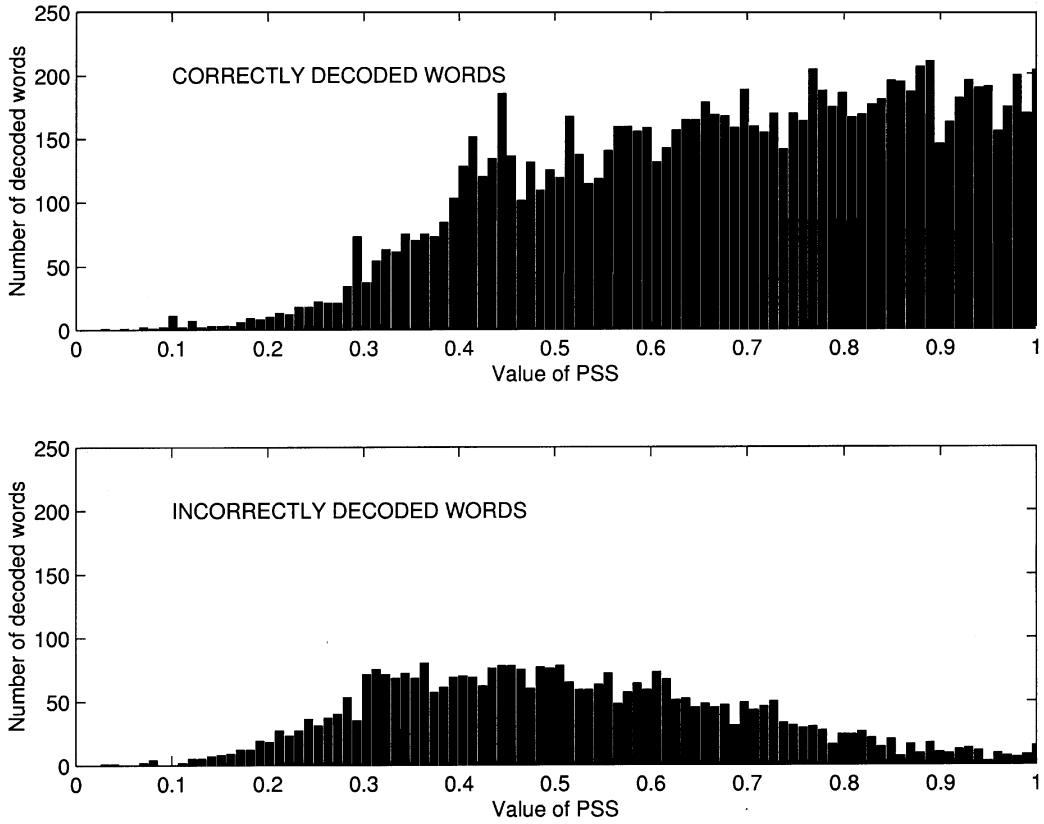

Fig. 4. Distributions of values of PSS for correct and incorrect words.

would be useful. We used the N-best CM (NB) as described in Section IV-B to provide this. If we assume that NB_i and PSS_i, the values of NB and PSS for word w_i, are independent estimates of the probability that word w_i is correct, the product of these values ($NBPSS_i$) gives a further estimate of this probability. Fig. 5 shows receiver operating curves formed from the three CMs NB, PSS and $NBPSS$ by varying the threshold above which words were classified as correct (N.B. this was run on *all* decoded utterances). Fig. 5 shows that PSS (squares) is generally a poor indicator of the status of a decoded word: if all decoded words are examined, its performance is no better than chance, but as the proportion of decoded words examined drops, its accuracy increases to close to 100%. NB (crosses) is better for high values of recall, but the maximum precision NB is capable of is 87.5% at a recall of about 50%. The single NB point to the left of this point is made from the values of NB that are exactly 1.0 (i.e., words that occur in all top 100 decodings) and the proportion of such words that are actually correct is about 85.5%. When PSS is combined with NB to form $NBPSS$ (asterisks), performance with high recall is slightly better than NB alone $NBPSS$ retains the ability of PSS to give high precision for low recall. This is useful if it is desired to identify a small number of decoded words as correct with a high confidence.

G. Discussion

Using information about how well a decoded word relates semantically to the other decoded words in an utterance can provide useful information about whether the word is correct. This information is largely independent of information derived from a "side information" based confidence measure, such as N-best and hence complements the latter. Although our original idea was that a semantically based CM would be able to identify incorrectly decoded words that were not cognate with the other decoded words in an utterance, we found unexpectedly that it was better at predicting correctly rather than incorrectly decoded words. The ability of the technique to identify correctly decoded words seems to be due to the presence of a set of commonly occurring content words in the WSJ data that are highly cognate (e.g., numbers, financial terms). The technique was capable of identifying incongruous words successfully (such as the example quoted in Section VI), but was unable to discriminate between these words and correctly decoded common words occurring in a previously unseen context. It was clear from the preliminary experiment that this approach would only ever be capable of giving confidence measures for infrequently occurring content words and future work will include integration of word probability measures with the semantic measures to enhance this ability.

VII. Summary and Discussion

Our motivation in this work was to develop techniques for word confidence estimation that are independent of the architecture and operation of the word recognizer. In doing so, we have attempted to take a broader view of the problem than previously and base our measures on additional sources of relatively independent information rather than on *ad hoc* "side information" derived from the recognizer's internal workings. We

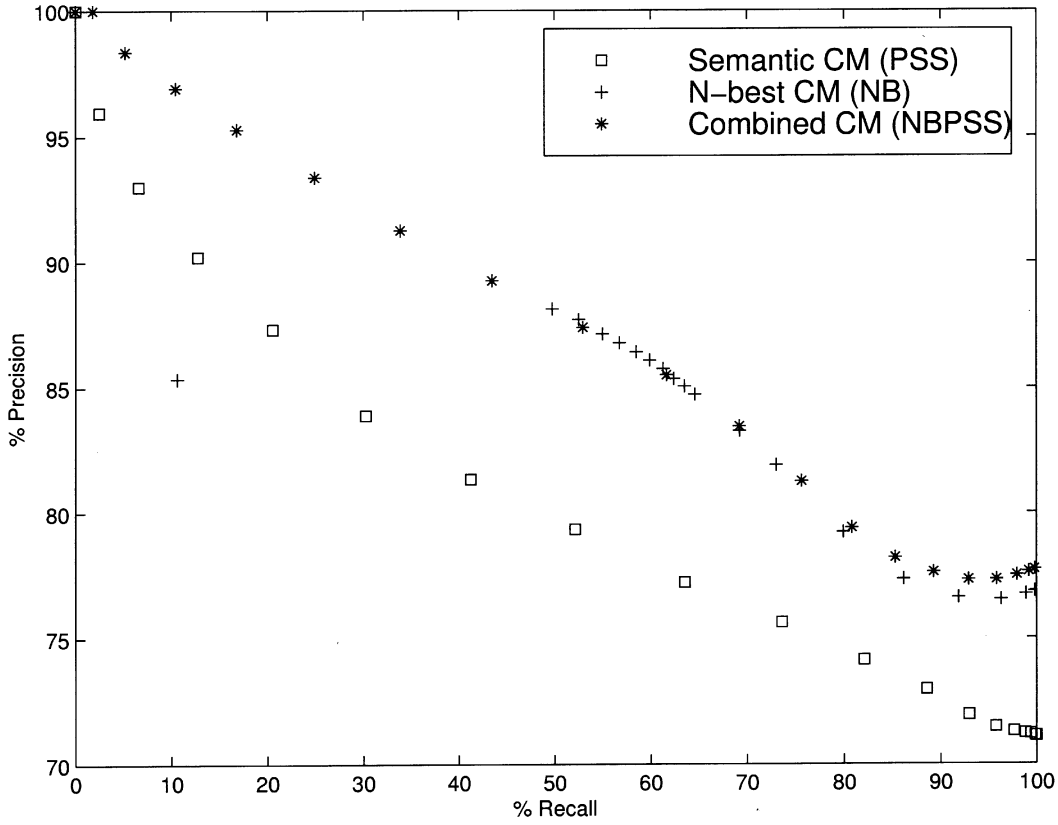

Fig. 5. Recall/precision curves for PSS, NB, and $NBPSS$.

TABLE VII
SUMMARY OF PERFORMANCE OF THE TECHNIQUES DISCUSSED IN THE PAPER

Technique	CER_{CM} (%)	CG (%)
N-best (baseline)*	24.0	23.5
Phone level + likelihood ratio (PL+LR)	29.2	5.8
Metamodels	21.9	31.1
LSA + NBest (NBPSS)	22.8	26.3

have examined ways of using two extra sources of information: a phone-loop recognizer working in parallel with the word recognizer and a "semantic distance" between the words present in the word recognizer output. Table VII summarizes the performance of the techniques discussed in the paper. In Section III, we developed some theory that gives an approach to using the output of a phone recognizer in tandem with the output from a word recognizer to derive confidence measures for the words output by the word recognizer. This theory was utilized in Section IV in a technique called "phone correlation." Phone correlation gives a small improvement in confidence over guessing, but Table VII shows that using a phone-loop recognizer to construct "metamodels" has much more benefit. Section V extended the ideas of Section III to develop the use of metamodels, in which putative word strings are constructed from the output of the phone recognizer. By comparing these strings with the word strings produced by the word recognizer a confidence measure can be produced. This is a more complex technique than either N-best or phone correlation that offers substantially better performance than both. It could also be used effectively in tandem with the N-best technique if N-best information is available. Metamodels attempt to capture the mapping from the output of a phone recognizer to the true transcription and as such may find application in other areas of speech recognition (in pronunciation modeling, for instance).

Section VI introduced an approach that was motivated by the idea that words decoded by the recognizer that are semantically "distant" from the other decoded words are more likely to be incorrect. The semantically based technique has the advantage of requiring no side information at all from the recognizer: on its own, it is a weak indicator of confidence, but when combined with N-best, it produced a very useful gain in the low recall/high precision region. N-best performs poorly in this region because there are a significant number of incorrect utterances that have an N-best "score" of 100%. We speculate that the semantically based technique would be most effective when used in an application that has several domains that are fairly independent of each other, but which uses a single vocabulary and language model.

We are currently in the process of testing the effectiveness of all these techniques and comparing them with conventional techniques, on a real dialogue system.

REFERENCES

[1] L. Chase, "Word and acoustic confidence annotation for large vocabulary speech recognition," in *Proc. 5th Eur. Conf. Speech Communication and Technology*, Sept. 1997, pp. 815–818.

[2] L. Gillick, Y. Ito, and J. Young, "A probabilistic approach to confidence estimation and evaluation," in *Proc. IEEE Conf. Acoustics, Speech, Signal-Processing*, Apr. 1997.
[3] T. Schaaf and T. Kemp, "Confidence measures for spontaneous speech recognition," in *Proc. IEEE Conf. Acoustics, Speech, Signal-Processing*, Apr. 1997.
[4] S. J. Cox and R. C. Rose, "Confidence measures for the SWITCHBOARD database," in *Proc. IEEE Conf. Acoustics, Speech, Signal-Processing*, 1996, pp. 511–515.
[5] D. D. Palmer and M. Ostendorf, "Improved word confidence estimation using long range features," in *Proc. 7th Eur. Conf. Speech Communication and Technology*, Sept. 2001.
[6] A. Asadi, R. Schwartz, and J. Makhoul, "Automatic detection of new words in a large vocabulary speech recognition system," in *Proc. IEEE Conf. Acoustics, Speech, Signal-Processing*, 1990, pp. 125–128.
[7] M. C. Benitez *et al.*, "Word verification using confidence measures in speech recognition," in *Proc. 5th Int. Conf. Speech Communication and Technology*, Nov. 1998, pp. 1082–1085.
[8] T. Robinson *et al.*, "WSJCAM0: A British English speech corpus for large vocabulary continuous speech recognition," in *Proc. IEEE Conf. Acoustics, Speech, Signal-Processing*, 1995, pp. 81–84.
[9] J. Jansen, J. Odell, D. Ollason, and P. Woodland, *The HTK Book*: Entropic Res. Labs., Inc., 1996.
[10] S. J. Cox and S. Dasmahapatra, "A high-level approach to confidence estimation in speech recognition," in *Proc. 6th Eur. Conf. Speech Communication and Technology*, Sept. 1999, pp. 41–44.
[11] S. Dasmahapatra and S. J. Cox, "Meta-models for confidence estimation in speech recognition," in *Proc. IEEE Conf. Acoustics, Speech, Signal-Processing*, June 2000.
[12] Z. Chair and P. Varshney, "Optimal data fusion in multiple sensor detection," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 22, no. 1, pp. 98–101, 1986.
[13] L. Chase, "Error-Responsive Feedback Mechanisms for Speech Recognizers," Ph.D., Carnegie Mellon Univ., Pittsburgh, PA, 1997.
[14] M. Weintraub *et al.*, "Neural-network based measures of confidence for word recognition," in *Proc. IEEE Conf. Acoustics, Speech, Signal Processing*, Apr. 1997.
[15] R. R. Sarukkai and D. H. Ballard, "Phonetic set indexing for fast lexical access," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 20, pp. 78–82, Jan. 1998.
[16] F. Wessel, S. Schlüter, K. Macherey, and H. Ney, "Confidence measures for large vocabulary speech recognition," *IEEE Trans. Speech Audio Processing*, vol. 9, pp. 288–298, Mar. 2001.
[17] R. Zhang and A. I. Rudnicky, "Word level confidence annotation using combinations of features," in *Proc. 7th Eur. Conf. Speech Communication and Technology*, Sept. 2001.
[18] C. Pao, P. Schmid, and J. Glass, "Confidence scoring for speech understanding systems," in *Proc. 5th Int. Conf. Spoken Language Processing*, Dec. 1998.
[19] J. R. Bellegarda, "A multispan language modeling framework for large vocabulary speech recognition," *IEEE Trans. Speech Audio Processing*, vol. 6, pp. 456–467, Sept. 1998.
[20] T. K. Landauer and S. T. Dumais, "A solution to Plato's problem: Representation of knowledge," *Psychol. Rev.*, vol. 104, pp. 211–240, 1997.
[21] C. D. Manning and H. Schutze, *Foundations of Statistical Natural Language Processing*. Cambridge, MA: MIT Press, 1999.

Stephen Cox (M'00) received the B.Sc. and M.Phil. degrees in physics and music from Reading University, Reading, U.K., and the Ph.D. degree in speech recognition from the University of East Anglia, Norfolk, U.K.

His research interests are in speech processing, especially recognition, and pattern recognition. He joined British Telecom Research Laboratories (BTRL) in 1984 where he was concerned with assessment of speech recognition algorithms. From 1987 to 1989, he worked on rapid speaker adaptation at the Speech Research Unit at DERA, Great Malvern, U.K. He returned to BTRL in 1989 as Head of a group developing robust speech recognition algorithms for use over the telephone network. He was appointed Lecturer in Electronic Systems Engineering at the University of East Anglia in 1991 and Senior Lecturer in 1998. From July to December 1994, he was a Visiting Scientist in the Speech Research Group at AT&T Bell Labs, Murray Hill, NJ, and during the autumn of 2000, was Visiting Scientist at Nuance Communications, Menlo Park, CA. He is the author of more than 50 papers in the fields of speech and pattern processing.

Dr. Cox is Chairman of the U.K. Institute of Acoustics Speech Group.

Srinandan Dasmahapatra received the B.Sc. (Hons.) degree in physics from St. Xavier's College, Calcutta University, India, and the Ph.D. degree in physics from the State University of New York at Stony Brook.

After postdoctoral fellowships in Trieste, Italy, and London, U.K., he switched from physics to research in speech recognition at the University of East Anglia, Norfolk, U.K., in 1998. He is currently a Lecturer in the Intelligence, Agents, and Multimedia Group, University of Southampton, Southampton, U.K., and is working on the Semantic Web and related issues within an EPSRC-funded IRC called Advanced Knowledge Technologies.