Automatic Call Routing with Multiple Language Models

Qiang Huang and Stephen Cox

School of Computing Sciences,
University of East Anglia,
Norwich NR4 7TJ, U.K.

{h.qiang|sjc}@cmp.uea.ac.uk

Abstract

Our motivation is to perform call routing of utterances without recourse to transcriptions of the training data, which are very expensive to obtain. We therefore use phonetic recognition of utterances and search for salient phonetic sequences within the decodings. An important issue in phonetic recognition is the language model. It has been demonstrated [1] that the use of an iterative language model gives benefits in speech recognition performance that are translated to improvements in utterance classification. However, an all-purpose language model sometimes produces decodings that are ambiguous, in that they apparently contain key phonetic sequences from several different routes, or non-informative, in that they apparently contain no useful phonetic sequences. This paper describes a method that uses multiple language models to detect useful information in such utterances. The outputs from recognizers that use these multiple models are examined by post-processing HMMs that decide whether putative sequences are present or not. It is found that using multiple language models increases performance significantly by classifying utterances that a single language model is unable to discriminate.

1. Introduction

Call routing refers to the technique of automatically relaying a customer's telephone enquiry to one of several appropriate destinations, using computational speech and language processing techniques. Transcribing calls for training purposes for a particular application requires considerable human effort, and it would be preferable for the system to learn routes without transcriptions being provided [2].

In this study, we assume that we are provided with a set of training utterances that have been labelled with their destination by an expert, but not transcribed into words or phonemes. We also assume (perhaps over-pessimistically) that we have no prior knowledge of the vocabulary or syntax of our application.

In this situation, one possible course of action is to use phone recognition and attempt to identify phonetic sequences that are salient to particular routes. Unfortunately, the speech signals are often of very poor quality, being subject to the usual distortion, bandwidth restriction and noise associated with telephone signals, and often compounded by the fact that callers usually speak casually and spontaneously, and sometimes with a strong accent.

Some approaches to the problem of extracting salient phonetic strings from these utterances are:

- Improve phone accuracy by using a variable length language model and building models for insertion and substitution; [3,4]
- Identify subword units (e.g. phonemes, phoneme strings, syllables and morphemes) from the recognised phonetic sequences by using clustering and segmentation methods; [5,6,7]
- Use matrix-based methods for classification, such as LSA, LDA, ICA, SVM, etc. [8,9,10]

Work at AT&T [1] showed that call routing performance using this phone-string utterance classification can be surprisingly close to what can be achieved by conventional methods involving word-trigram language models that require manual transcription. The method described in [1] combines automatic training of application-specific phonotactic language models together with token sequence classifiers.

Our own experiments, using data different from that used by AT&T, showed that this technique gave only a small benefit in phone recognition accuracy, but was useful for finding salient phoneme strings. However, we found that, in some cases, it was impossible to obtain salient phoneme sequences from the recognised utterances even when it was known that they occurred within the utterance. The reason may be that when building a single language model with the collected utterances from all call routes, the salience of a particular sequence for a particular route is lost in the "noise" from mis-recognised sequences of phonemes from the other routes. Hence we sought a way of making the language model more sensitive to the keywords occurring in the utterances. In our system, an independent corpus is used to build an n-gram phonotactic language model that enables an initial recogniser to be built to decode all the training utterances. This model is refined iteratively using the output from the recogniser as the basis for the next language model. A specific language model for each call route is then built using the utterances from this call route. These are much more sensitive to key salient phoneme sequences in the utterance.

The structure of the paper is as follows: in section 2, the data corpus used is introduced. Section 3 describes in detail the language modelling techniques, section 4 presents experiments and analysis of results, and we end with a Discussion in section 5.

2. Database

The application studied here was the enquiry-point for the store card for a large retail store. Customers were invited to call up the system and to make the kind of enquiry they would normally make when talking to an operator. Their calls were routed to 61 different destinations, although some destinations were used very infrequently. 15 000 utterances were available, and a subset of 4511 utterances was used for training and 3518 for testing, in which 18 different call types were represented. Some of these call types are quite easily confused e.g. PaymentDue and PaymentDate, PaymentAddress and Changeaddress. Phoneme recognition of the input speech queries was performed using an HMM recogniser whose acoustic models had been trained on a large corpus of telephone speech and which had separate models for males and females. The average length of an utterance is 8.36 words. In addition, transcriptions of the prompts from the Wall Street Journal (WSJ) database were used to generate phoneme-level statistical language models for initial training. These models were generated using a scheme for backing off to probability estimates for shorter n-grams.

The size of the vocabulary is 1208 words. To get a feel for the difficulty of the task, the mutual information (MI) between each word and the classes was calculated. By setting a threshold on this figure, we observed that there were about 51 keywords occurring in 4328 utterances which were capable on their own of classifying a call with high accuracy (some utterances had no keywords).

3. Modelling

3.1. Model Structure

Figure 1 shows the method used to produce an initial language model.

The algorithm follows that described in [1]:

1. Build an n-gram language model (LM) using the dictionary transcriptions of the WSJ corpus (we used n=6). Make this the current LM.
2. Use the current LM in the recognizer to produce a set of phone strings.
3. Build a new LM based on the recognizer phone strings:
4. If `niterations <=threshold`, `goto 2` else `finish` and produce a single language model for all routes.

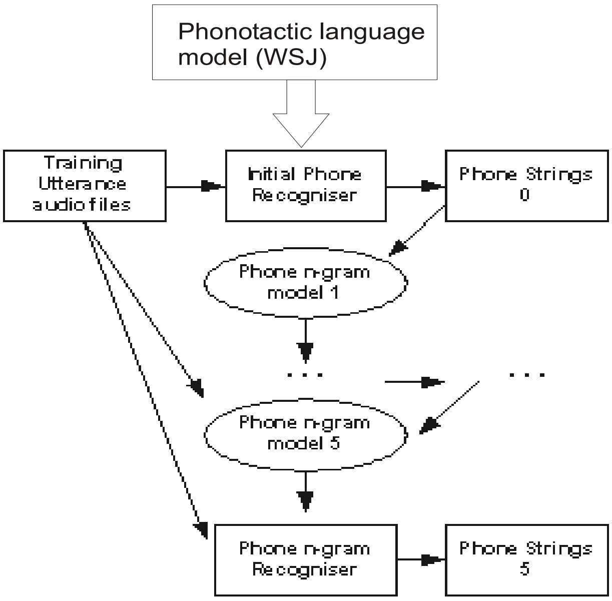

Figure. 1 The Iterative training procedure

The phone strings are now segmented and clustered so that salient phone sequences for each route can be identified. This is done as follows:

FOR EACH ROUTE

1. Segment each recognized phone string in the route into all possible sequences of 3,4, … , 9 phones.
2. Estimate the MI for each sequence, and identify the salient sequences as the sequences with the highest MI [11].
3. Cluster the salient sequences within the route. This is done by calculating and combining two measures of distance (using dynamic programming techniques) for each pair of sequences:
   - The Levensthein distance between the phone symbols representing the sequences.

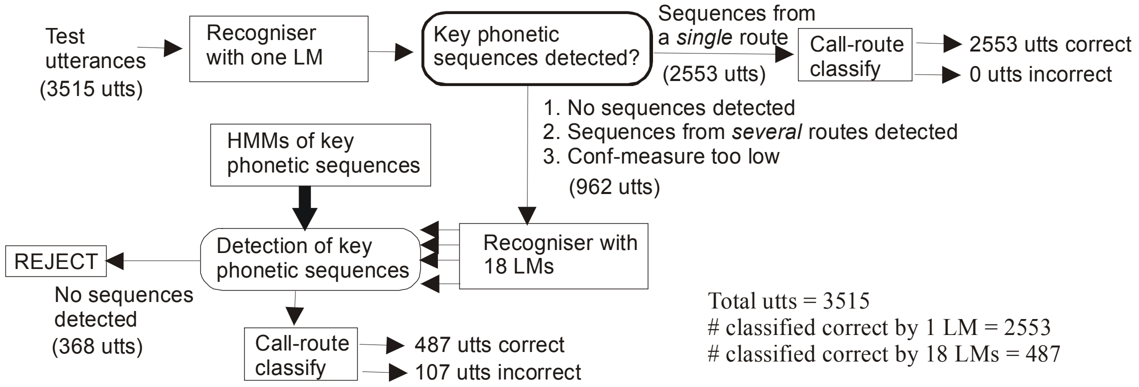

Figure 2: The Recognition Process

- The acoustic distance in "MFCC space" between the two waveform segments representing the sequences.

4. Use a simple lexicon pruning scheme that eliminates long agglomerations of short primitives [12].

At this point, we have generated a set of clustered phone sequences for each route. Each phone sequence corresponds to a sequence of frames, and the frame sequences within a cluster are used to build an HMM These HMMs are used later to estimate the class of a segment output by the recognizer (see section 3.2).
Finally, we build a language model for each route, as follows by collecting together the recognised phonetic sequences of utterances from each route and using them to construct a language model.

After iterating the LM, detection of key phonetic sequences improves. However, many utterances do not produce any sequences or produce several sequences from different routes. For recognition, we use a "divide and conquer" approach. Utterances that yield one or more sequences from the same route are classified immediately as that route, and utterances whose output is ambiguous, in that they yield no sequences, or sequences from several routes, or whose recognition confidence is too low to trust, are subject to a more detailed recognition pass in which separate LMs for each route are used. This has the advantage of only applying the extra computational effort required to use multiple LMs for those utterances that need this. In practice, if lattices are used, the additional computational effort is not too great. The confidence measure used was the measure available from the Nuance speech recognizer v8.0.

Hence recognition proceeds as follows.

1. A single language model is used in the recognizer to produce an output phone string.
2. Any phonetic sequences in the output string that also occur within any of the clusters of key phonetic sequences in any of the routes are found.
3. ***IF** the number of key phonetic sequences found is one or more **AND** the sequences all belong to the same route:*
   the utterance is classified as belonging to this route.
   ***ELSEIF** the number of key phonetic sequences is zero **OR** there are one or more sequences from different routes **OR** the confidence measure of the whole utterance is lower than some threshold*:
   the utterance is re-recognized using all 18 language models.
4. Recognition using multiple language models works as follows. 18 recognized phonetic sequences are output, one from each recognizer (as shown in Figure 2), and key phonetic sequences are detected in each output.

   IF there are one or more sequences from different routes:
   Putative sections of the speech that contain keywords are identified by comparing the symbolic output of a recognizer using a certain LM with the sequences that were used to form the HMMs of the clustered key phonetic sequences for this LM. These HMMs are then used to determine the likelihood of each sequence given the output string, and the utterance is assigned to the route of the highest likelihood.
   ***ELSEIF** the number of key phonetic sequences is zero*
   The utterance is not classified (rejected).

Call type classification is done using a vector-based approach as described in [8]. It is perhaps surprising that this classifier gets 100% accuracy (2553/2553) on utterances in which all the sequences are apparently from the same route—we attribute this to the fact that the 18 call-types were used were highly independent in their use of keywords.

Figure 2 gives an overview of the whole process, together the number of utterances that were involved in each stage.

3.2. Key Phonetic Sequence Detection

Key phonetic sequences can be incorrectly matched to incorrect segments of the utterance, causing false alarms. To combat this problem, we use matching in the acoustic domain as well as the symbolic domain. HMMs for 41 key phonetic sequences whose number of occurrences was larger than a threshold (we used 30) were built. Each key phonetic sequence was modelled by a five-state left-to-right HMM with no skips and each state is characterised by a mixture Gaussian state observation density. A maximum of 3 mixture components per state is used. The Baum-Welch algorithm is then used to estimate the parameters of the Gaussian densities for all states of subword HMM's.

We use key phrase detection as described in [13][14]. By using the phonetic output from the recogniser, the position in the utterance waveform of putative strings can be identified, and this section of the waveform is input into the phonetic sequence HMMs. Detection of phrases is achieved by monitoring the forward probability of the data given the model at any time and searching for peaks in the probability. If full-likelihood recognition is used, we estimate the score $S_f(w,t)$:

$$S_f(w,t) = \frac{\alpha(e_w,t)}{\sum_s \alpha(s,t)} \qquad (1)$$

In equation (1), $S_f(w,t)$ is the forward probability of word w at time t [13]. In practice, we used the Viterbi equivalent of equation (1) to determine the likelihood.

4. Experiments

4.1. Phone accuracy based on one LM

Figure 3 illustrates the effects of

(a) using the recogniser output strings to construct a new language model as described in section 3.1;

(b) using 18 different LMs as well as a single LM.

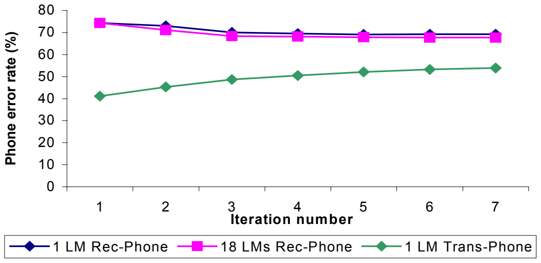

Fig 3. Phone error rate using 1 LM and 18 LMs

Rec-Phone: *Build language model using recognised phonetic sequences of utterances from training set;*
Trans-Phone: *Build language model using phoneme transcriptions of words of utterances from training set*
1 LM: Recognition using one language model;
18LMs: Recognition using 18 language models.

Figure 3 shows that the phone error rate is very much higher when recognised phone sequences (Rec-Phone) rather than dictionary transcriptions (Trans-Phone) are used to build an LM. However, an interesting point is that iterative performance *decreases* when the transcriptions are used, but *increases* when the recognised strings are used. This is probably because, when the recognised strings are used, the initial LM, which is trained on WSJ, does not reflect the distribution of n-grams in the data, and so performance is poor. However, the vocabulary in the data is quite small, so that after even a single recognition pass, although the error-rate is high, the new LM is a better reflection of the n-grams in the data. This has the effect of improving the phone recognition performance, and this improvement continues with each iteration.

When we use an initial language model built using dictionary phoneme transcriptions, the performance is initially much better than using an LM trained on an independent corpus, as would be expected. However, because of the small vocabulary size and the relatively high number of occurrences of a few phonetic sequences, any errors in recognition of these sequences dominate, and this leads to an increasing overall error-rate.

These results are not as good as those obtained by Hiyan [1] using an iterative language model. This may be because of the difference in the speech recognisers, or, more likely, in the average length of the phrases in the different vocabularies, which are much shorter than the phrases used here.

4.2. Classification Accuracy

Iteration No.	1	2	3	4	5
Phone accuracy	25.7	27.1	30.0	30.6	31.0
Classif-ication accuracy Rec-Phone (%)	44.3	60.4	69.4	72.1	72.6

Table 1. Phone recognition accuracy and call routing accuracy

Table 1 shows the call-routing classification performance when a single LM is used and the LM is iterated. What is interesting here is that an apparently small increase in phone accuracy on iteration gives rise to a huge increase in call-routing accuracy. This is because although the overall phone error-rate improves only slightly, the error rate on the key phonetic sequences is greatly improved, leading to improved classification performance. Note that performance on this dataset when the dictionary translations of the transcriptions of the utterances are used is 93.7%.

Name	Trans-Phone	1 LM	1 LM + Multiple LMs
Correct classification rate (%)	93.7	72.6	86.5

Table 2. Comparison of correct classification rate
Trans-Phone: language model built with dictionary phoneme transcriptions of the utterances;
1 LM: iterative language model built using recognition output;
1 LM + Multiple LMs: Using the two-pass approach described in section 3.1.

Table 2 compares the call-routing classification accuracies. The accuracy achieved using the two pass system with multiple LMs (86.5%) is much better than that using a single iterated LM, but not quite as good as that obtained by using the dictionary transcriptions.

It could be argued that it is not possible to say whether the improvement shown in column 4 of Table 2 compared with column 3 is due to the use of multiple LMs or to the use of the HMM post-processor. However, when a single LM is used, the situation is either that there are one or more fairly unambiguous output sequences from a single call type, or there are many noisy and ambiguous sequences whose positions are not well-defined. It is very difficult to process these putative sequences with all the HMMs of key phonetic sequences. Using multiple LMs has the effect of producing relatively unambiguous sequences from only a small subset set of call-types, whose position in the waveform is quite well-defined. This reduces the number of HMM sequences that need to used and hence also the difficulty of application.

5. Discussion

In this paper, we have presented a method for automatic call routing in which we do not require transcriptions of the training utterances, only the route of each utterance. The technique is based on phonetic recognition of utterances, and we have focused on the design of the language model in this recognition process. Our conclusions are that iterating a single phone language model (as described in [1]) is highly beneficial to performance, but performance can be further increased by using multiple language models for recognition for utterances whose content is ambiguous when a single language model is used. Using multiple LMs inevitably gives rise to identification of false keywords, but this difficulty is resolved by the use of post-processing HMMs which estimate the likelihood of the putative keyword phonetic sequence being present in the waveform. Future work will concentrate on use of confidence measures and classification of ambiguous utterances. We will also investigate the use of “lightly supervised” adaptation, in which a small proportion of the utterances available have been transcribed [15].

6. Acknowledgment

We are grateful to Nuance Communications for providing the data for this study.

7. References

[1] Hiyan Alshawi, “Effective Utterance Classification with Unsupervised Phonotactic Models”, in Proc. HLT-NAACL 2003, pp. 1-7, Edmonton.

[2] Qiang Huang, Stephen Cox, “Automatic Call-routing without Transcriptions”, in Proc. EuroSpeech, Geneva, 2003.

[3] Deligne S., Bimbot F., “Inference of Variable-length Linguistic and Acoustic Units by Multigrams.” Speech Communication 23, pp. 223-241, 1997.

[4] Thomas Hain, Philip C. Woodland, “Modelling Sub-Phone Insertions and Deletions in Continuous Speech”, in Proc. International Conference on Spoken Language Processing 2000, Beijing, China

[5] K. Ng, V.W. Zue, “Subword Unit representations for spoken document retrieval”, in Proc. Eurospeech 1997.

[6] T. Nagarajan, Hema Murthy, “Segmentation of Speech into Syllable-like units”, in Proc. EuroSpeech 2003, Geneva.

[7] D. Petrovska-Delacretaz, A. L. Gorin, J. H. Wright, and G. Riccardi. “Detecting Acoustic Morphemes in Lattices for Spoken Language Understanding”, in Proc. International Conference on Spoken Language Processing 2000, Beijing, China.

[8] S. Cox. , “Discriminative Techniques in Call Routing”, Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003, HongKong, China

[9] Leopold, E.and J. Kindermann, “Text Categorization with Support Vector Machines. How to Represent Texts in Input Space?” Machine Learning, 2002. 46: pp.423-444.

[10] Lee, T.-W., Lewicki, M.S., and Sejnowski, T.J., “ICA Mixture Models for Unsupervised Classification and Automatic Context Switching.” Proc. International Workshop on Independent Component Analysis , 1999.

[11] S. Cox and B. Shahshahani, “A Comparison of some Different Techniques for Vector Based Call-Routing” Proc. Workshop on Innovation in Speech Processing, Stratford, April 2001.

[12] Fuchun Peng, Dale Schuurmans, “A Hierarchical EM Approach to Word Segmentation”, in Proceeding of the Sixth Natural Language Language Processing Pacific Rim Symposium. Nov. 2001, Tokyo, Japan

[13] J. R. Rohlicek, P. Jeanrenaud, K. Ng, H. Gish, B. Musicus M. Siu, “Phonetic Training and Language Modeling for Word Spotting”, in Proc. of IEEE International Conference on Acoustic, Speech, and Signal Processing, 1993.

[14] Tatsuya Kawahara, Chin-Hui Lee, Bijing-Hwang Juang, “Flexible Speech Understanding Based on Combined Key-Phrase Detection and Verification”, IEEE Transactions on Speech and Audio Processing, Vol.6, No. 6, November 1998.

[15] D. Giuliani and M. Federico, "Unsupervised Language and Acoustic Model Adaptation for Cross Domain Portability", in Proc. ISCA ITR Workshop, Sophia-Antipolis, France, 2001.