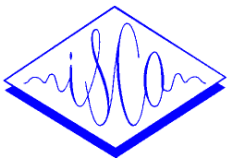

6th International Conference on Spoken Language Processing (ICSLP 2000)
Beijing, China
October 16-20, 2000

# IMPROVED VARIABLE PRESELECTION LIST LENGTH ESTIMATION USING NNs IN A LARGE VOCABULARY TELEPHONE SPEECH RECOGNITION SYSTEM

*J. Macías-Guarasa, J. Ferreiros, J. Colás, A. Gallardo-Antolín and J. M. Pardo*

Grupo de Tecnología del Habla. Departamento Ingeniería Electrónica. Universidad Politécnica de Madrid.
E.T.S.I. de Telecomunicación, Ciudad Universitaria s/n, 28040, Madrid, Spain
e-mail: {macias, jfl, colas, gallardo, pardo)@die.upm.es

10.21437/ICSLP.2000-396

## ABSTRACT

In very large vocabulary hypothesis-verification systems, the fine acoustic matcher is usually the most time consuming, so that the main concern is reducing the preselection list length as much as possible. Traditionally, these systems use a too high fixed preselection list length, increasing computational demands over the really needed.

The idea we are proposing is estimating a different preselection list length for every utterance, so that we can lower the average computational effort needed for the recognition process. As we will show, it's even possible that the resulting system outperforms the fixed length one in error rate, even when reducing computational cost.

This paper presents a detailed study on a NN based approach to variable preselection list length estimation. The main achievement has been a relative decrease in error rate of up to 40%, while getting a relative decrease in average preselection list length of up to 31%.

## 1. INTRODUCTION

Computational demands are one of the main factors to take into account when designing systems supposed to operate in real-time, specially when talking about public information services using the telephone network.

Telephone information service providers are demanding systems and algorithms that allow them to increase the number of active recognizers to run in dedicated hardware, to be able to significantly decrease production costs.

According to this scenery, systems based on the hypothesis-verification paradigm are generally used, so that the output of a rough analysis module, with low computational demands, is fed to a detailed matching module [1][2][5]. The first one generates a list of candidate words, known as the preselection list, which will be further processed with a much more detailed strategy. For the whole system to be successful, the rough analysis module must ensure that the right word is within the list it generates with high probability, so as not to degrade the overall performance.

In hypothesis-verification systems, the fine acoustic matcher is usually the most time consuming, so that the main concern is reducing the preselection list length as much as possible. This is not an easy task, especially when low detailed acoustic models are used in the preselection stage. Traditionally, these systems use a fixed preselection list length, estimated according to the results obtained during system development so that a minimum error rate is achieved.

Using this approach, designers are obliged to use a high number of words to include in the constant length preselection list. Our proposal is making this list length variable, different for every utterance, depending on any know-in-advance system parameter.

The key factor to evaluate different methods is calculating the average preselection list length (average effort) while keeping the required error rate. The idea in the optimal case is estimating the minimum number of words we must pass to the verification stage so that we keep the error rate under a predefined maximum, thus reducing the average preselection list length and therefore the computational demands of the verification stage.

In the past we have developed several approaches facing the estimation of preselection list lengths as opposed to using fixed length lists, with promising results [2]. In [3] we attempted for the first time the use of NNs for this purpose in a limited task. NNs are one of the most suitable strategies to cope with the proposed problem, given the lack of explicit and implicit a priori information on the estimation task. In this paper we refine and extend the NN approach leading to working systems that performs better than the fixed length based traditional systems on a much more complex task, using significantly bigger databases.

## 2. SYSTEM OVERVIEW

The general system architecture is shown in Figure 1. The current version of the hypothesis module follows a bottom-up, two stage strategy, as shown in Figure 1. The main preselection modules are detailed in [4]. An estimator module is in charge of deciding the length of the list of words to be passed on to the detailed analysis module, using for that purpose a certain set of parameters extracted from the front-end and rough analysis processes. This estimator module will be based on a NN.

In general, and given the proposed task, it is clear that computational requirements are closely related to different factors: modeling complexity and search effort in the preselection and verification algorithms. Fixing those, it is intuitive that as we lower the computational requirements (lowering the average preselection list length) we will get a decrease in recognition rate. Nevertheless, it's also posible that the opposite effect happens: increasing inclusion rate while lowering search effort. So, we face a complex optimization problem, trying to find a balance between computing requirements and recognition accuracy. Anyway, our priority will be keeping a certain error rate, **under 2% in our case**.

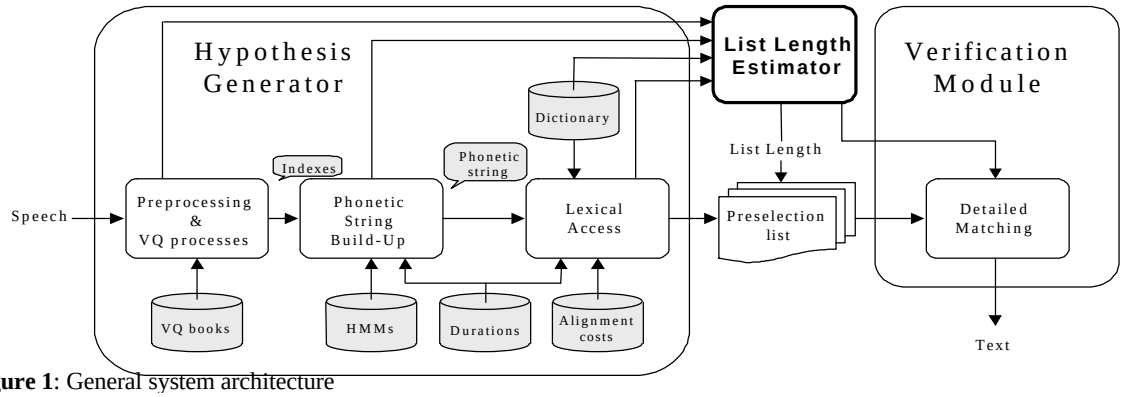


**Figure 1**: General system architecture

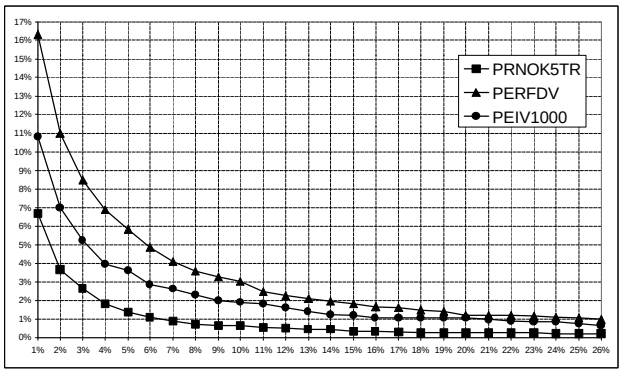


**Figure 2**: Inclusion error rate curves for the baseline experiment

## 3. DATABASES AND DICTIONARIES

In our experiments we have used part of VESTEL, a realistic telephone speech database, captured using the Spanish PSTN [6]. The subset that we have used is divided in three parts:

- PRNOK5TR: Devoted to generic system training, is composed of 5820 files (3011 different speakers)
- PERFDV: Devoted to testing and originally designed to make "vocabulary dependent tests". It's composed of 2536 different utterances (2255 different speakers).
- PEIV1000: Devoted to testing and originally designed to make "vocabulary independent tests". It's composed of 1434 utterances (1351 different speakers)

In the tasks described in this paper, we have used dictionaries composed of 10000 words.

## 4. BASELINE FIXED LENGTH LIST SYSTEM

In the preselection stage we use semicontinuous HMMs, due to their good compromise between modeling accuracy and computational requirements. The unit inventory is composed of 23 automatically clustered models, plus two additional ones for initial and final silence.

The baseline experiment used fixed length lists and we calculated the inclusion rate achieved for every possible length in the preselection list. In Figure 2 we show the preselection error rate curves as a function of preselection list length (measured as a percentage of dictionary size). As can be seen, for PEIV1000 we achieve 2% error rate for around 10% of dictionary size, that is around 1000 candidates (916 exactly). PERFDV needs 1353 candidates (~13%) to get the same results.

So, our target will be achieving **at least** the same performance using variable preselection list lengths, estimated using a NN.

## 5. NNs AS VARIABLE PRESELECTION LIST LENGTH ESTIMATORS

In our case, we opted to use NNs given the characteristics of the system we are developing: the absence of explicit knowledge on the relationship (if it exists) between the parameters we can use and the length we want to estimate.

### 5.1. Network topology

We use a simple MLP with a single hidden layer testing a variable number of hidden units.

### 5.2. Parameter inventory and input layer coding

We have selected an inventory of 33 parameters. They can be divided in: *direct parameters* (directly obtained from the acoustic utterance or the preselection process), *derived parameters* (applying different types of normalization), and *lexical access statistical parameters* (calculated over the lexical access costs distribution, for the best scored words and using different number of elements in the calculation).

The input parameters can be coded using different coding schemes and different number of neurons:

- For the single input neuron per parameter case, we have tested the following coding schemes: *no coding*; *linear scaling*; *simple normalization* to a normal distribution with $\mu=0$ and $\sigma=1$; and *simple normalization with clipping*
- For the multiple input neuron per parameter case: *Linear or non-linear mapping*; *thermometric or single neuron activation* and, finally, *floating or fixed output coding*.

### 5.3. Output coding

The output coding strategy is fully detailed in [3] and we will only refer here to the main ideas:

- Every output neuron is trained to indicate a different preselection list length

- The list length limits that every output neuron represents are trained with a criterion that aims to get, when possible, a uniform number of training samples for all of them

## 5.4. NN output postprocessing

The NN output values are finally post-processed to obtain the final hypothesis list length. The idea is further increasing the proposed length, so that mismatches between the training and testing sets are compensated to a certain extent.

The two alternatives tested when deciding the final preselection list length are (full details in [3]):

- The winner output neuron decides (called WN method).
- The length is calculated as a linear combination of normalised activations multiplied by upper limit of the corresponding segment, as follows (from now on LC):

$$length = \sum_{i=1}^{NumOutputNeurons} Neuron_{length}(i) \cdot normact(i)$$

$Neuron_{length}$ is the upper limit of the corresponding output neuron and *normact* is the normalised activation of this neuron.

In both cases, an additional fixed or proportional threshold can be added to produce the list length to be finally used. Of course, these thresholds are also obtained during the training phase, imposing the achievement of a predefined error rate, higher than the target system one.

## 5.5. Parameters, topology and coding scheme selection

Preliminary experiments gave enough evidence of the usefulness of the proposed methods [3]. The next step was dealing with bigger databases to confirm the good tendency showed but, first of all, we needed a clear and objective estimation on the discriminative power of every parameter and the influence of different coding methods and topologies in the results obtained.

So, we decided to propose a less ambitious task so that we could more easily get an homogeneous training set. The approach we tested was designing a discriminative network trained to distinguish whether a certain word had been recognized in the first position of the list or not, simplifying the discrimination task.

The assumption in this case is that a good behavior in the simple discrimination task will lead to good results in the list length estimation process also.

We launched experiments for all 33 available parameters, using all available coding techniques and topology alternatives, which sums up to almost 2800 experiments with their corresponding results. The conclusions we obtained can be summarized as follows:

- The discrimination results have outperformed our expectations. The maximum rates achieved are around 70-75%, which is reasonably good taking into account the simplicity of the proposed discrimination system
- There are not statistical differences when comparing different methods using the same parameter, showing that the main factor is the quality of the parameter itself. So, we decided to use the simplest network topology, with one input neuron and coding the parameters using simple normalization
- The set of parameters with higher discrimination power is consistent across all databases, confirming the fact that the network is actually able to extract the discrimination information it needs to perform reasonably well.
- The best parameter in all the experiments has been the standard deviation of the lexical access costs, measured over the list of the first 10 candidates in the preselection list. This confirms experimental evidence we had in our Group related to the relationship of dispersion measures in acoustical and lexical costs and the recognition confidence.

## 5.6. Building the final parameter inventory

After studying the relative performance of all the available parameters we faced the task of improving the result by combining them. Our strategy for building up the final parameter inventory was as follows:

- We started the inventory with the parameter with higher discrimination rate
- We run experiments combining the best parameter with all the others in the top ranked list
- We completed the inventory adding all parameters that showed the highest relative increase in discrimination power when being used along with the best one.

So, we finally came up with an inventory of 8 parameters that outperforms the single input parameter case.

# 6. EXPERIMENTAL RESULTS

## 6.1. Final topology

In the NN list length estimator, we used the 8 input parameters discussed above, coded using simple normalization, 5 neurons in the hidden layer, and 10 output neurons. The neuron outputs were assigned list lengths using the approach described in [3].

## 6.2. Comparison strategy

Selecting a suitable comparison strategy was one of the most challenging aspects of our study. As the system demonstrated to be able to improve simultaneously in error rate and computing demands, our comparison strategy is oriented to evaluating both improvements at the same time.

If we consider the NN estimation procedure we use, it's clear that in every case we will get a measure of preselection rate and average effort associated to it, that is, a single point in the space [averageEffort x inclusionRate]. Our baseline experiment with fixed length lists offers us a complete curve (the ones shown in Figure 2), although we established as the working point the one corresponding to an inclusion rate of 98%.

When comparing both approaches (fixed vs. variable list length), we could just use the reference of both isolated points.

If the NN approach obtains a higher rate and an average effort lower than the fixed list length approach, we could conclude the NN clearly outperforms the other system. This simple comparison approach has two main problems: First of all, if we get a lower rate but a lower average effort, the comparison is not so clear. Second, we don't have any information on the sensitivity of the results to variations on the estimated thresholds and the network parameters, being difficult to decide whether the data obtained is following a uniform tendency of improvement. So, it's of outmost importance to extend the analysis so that we have a wider perspective on the comparison.

If we recall the use of thresholds to further increase the network estimation, we can clearly see that their use implies a change in the average effort and, in general, a change in inclusion rate. In every experiment we get a single threshold value but, if we vary this value in a certain range, we could obtain successive values (averageEffort x inclusionRate) that could be considered similar to the building of a preselection rate curve, comparable to the one we obtained with fixed length lists. The comparison we will make will be based in the verification of the improvement of the NN approach over the range of interest of the threshold variation, both in error rate and computational effort. By design, this threshold will be the one needed to obtain rates varying from 96.5% to 99% (around our target: 98%).

Anyway, we still need relative quality measurements related to the working points. We calculate the relative decrease in preselection error rate provided the preselection fixed list length equals the average effort in the NN system; and the relative decrease in average effort provided the inclusion rates are the same.

## 6.3. Results

After running the selected experiments, the general conclusions we extracted are:

- The experiments based in the WN methods are unable to achieve good results, due to the lack of precision in the discrimination process when we extend the task to more than two cases.
- The systems using the LC method and fixed threshold show much better results, clearly outperforming the fixed length approach in all the experiments.

Studying all the results using the LC method and fixed threshold, the following observations arise:

- In all the experiments, the NN approach got improvements both in rate and effort, being consistent along the full range of values under study
- The relative error rate improvements have an average value around 7-8% for PERFDV and 18-28% for PRNOK5TR and PEIV1000.
- In the best experiment for the PEIV1000 dababase, we have got a relative improvement of up 40.74% in inclusion error rate with around a 31% decrease in average effort. For the PERFDV database, a 10% improvement in error rate was achieved while getting a 13% improve in average effort.
- For the PERFDV we have also got the 2% error rate objective, but with a small increase of average effort over the 10% of dictionary size. It is explained considering the higher difficulty of this list, as shown in the curve results of Figure 2.

Due to the database sizes used, we could not ensure the statisticall significance of the differences with the fixed list length approach. Nevertheless, we firmly believe that the consistency of the improvements along the full range of study in both rate and effort, validates the NN approach

# 7. CONCLUSIONS AND FUTURE WORK

The main conclusions of the presented paper are:

- The use of neural networks as preselection list length estimators has proven to be an excellent alternative to the traditional approach of using fixed length lists, outperforming it for a wide range of operation conditions.
- The best parameters according to discrimination performance are the ones related to the standard deviation of the lexical access costs, calculated for a short number of candidates in the preselection list

The main task to face in a near future is switching to other tasks with bigger databases on which the statistical significance tests can lead to definitively concluding results. Additionally, one of our main concerns is the extensibility of the proposed approaches to speech recognition systems having a better performance than the one presented.

# 8. REFERENCES


1. Macias-Guarasa, J., Gallardo, A., Ferreiros, J., Pardo, J. M. and Villarrubia, L. "Initial Evaluation of a Preselection Module for a Flexible Large Vocabulary Speech Recognition System in Telephone Environment". ICSLP 96.
2. Ferreiros, J., Macías-Guarasa, J., Gallardo-Antolín, A., Cordoba, R., Pardo, J. M. and Villarrubia, L. "Recent Work on a Preselection Module for a Flexible Large Vocabulary Speech Recognition System in Telephone Environment". ICSLP 98
3. Macías-Guarasa, J., Ferreiros, J., Gallardo, A., San-Segundo, R., Pardo, J.M. and Villarrubia, L. "Variable Preselection List Length Estimation Using Neural Networks in a Telephone Speech Hypothesis-Verification System". EUROSPEECH'99
4. A. Gallardo-Antolín*, J. Ferreiros, J. Macías-Guarasa, R. de Córdoba and J. M. Pardo. "Incorporating multiple-HMM Acoustic Modeling in a Modular Large Vocabulary Speech Recognition System in Telephone Environment". In this conference proceedings.
5. Ney, H. and Billi, R.. "Prototype Systems for Large Vocabulary Speech Recognition: Polyglot and Spicos". EUROSPEECH'91
6. Tapias, D., Acero, A., Esteve, J., and Torrecilla, J.C. "The VESTEL Telephone Speech Database". ICSLP 94