Artificial Neural Networks in Recognition of Phonetic Features in Speech

Todd A. Stephenson
Centre for Cognitive Science, University of Edinburgh
2 Buccleuch Place, Edinburgh EH8 9LW, Scotland, UK
E-mail: *Todd.Stephenson@ed.ac.uk*
Phone: +44 131 650 4414; Fax +44 131 650 6626

April 17, 1998

Abstract

A set of recurrent artificial neural networks are used for speech recognition. By representing speech waves as vectors of mel-cepstral coefficients and energy, we can train a neural network to classify the values of phonetic features in a given sentence of speech. The effectiveness of the training depends both on the range of possible values for a given feature classification, on the distribution of the values in training samples, and possibly on how well a given feature is represented by cepstral analysis. The nets are able to identify both specific feature values and broader feature value groupings in speech.

1 Introduction

A popular method for doing speech recognition is to use hidden Markov models (HMM's). HMM's approach speech recognition is to determine the probability of the speech being a certain utterance. Meanwhile, neural networks take a non-probabilistic approach to problem solving. They can be used to learn the patterns in the input that give the desired output and to generalise characteristics of the input.

In either approach to speech recognition we start with a digital waveform. This, however, is not a good form to build a speech recogniser with. Mel frequency cepstral analysis (Owens, 1993) provides an effective method of converting a digital signal into vectors of coefficients that describe the speech. Using HTK (Young *et al.*, 1996), the speech signals are converted into speech vectors of 13 elements: 12 are the cepstral coefficients and the 13^{th} is the energy. Each vector deals with a moving window of 25 ms. Each succeeding vector shifts the window 10 ms, giving an overlap of 15 ms with the previous window. In this paper, I will be referring to these each vectors as frames.

Now, once the speech is broken down into these frames, we need to determine what about the speech we want to be able to recognise. Possible examples are whole words, syllables, phonemes, or phonetic features. It is this last one, phonetic features, that the networks will be trained to recognise in given speech.

2 The Models

The NICO toolkit (Ström, 1997) is used to build the networks. Six different networks were built, each to learn a different feature. The chosen features for this network are: centrality, frontback, manner, phonation, place, and roundness. Depending on the network, there were from three to nine output values for a network (Table 1). This is based upon the work done in Kirchhoff (1996). For example, the phoneme [d] is classified as [CENTRALITY: `nil`, FRONTBACK: `front`, MANNER:

CENTRALITY	FRONTBACK	MANNER	PHONATION	PLACE	ROUNDNESS
central(6)	back(23)	vowel(37)	voiced(61)	low(7)	unrounded(79)
full(34)	front(62)	fricative(18)	unvoiced(27)	medium(12)	rounded(9)
nil(49)	nil(3)	approximant(9)	silence(12)	high(18)	silence(12)
silence(12)	silence(12)	nasal(7)		labial(8)	
		occlusive(18)		dental(5)	
		silence(12)		coronal(26)	
				velar(10)	
				glottal(3)	
				silence(12)	

Table 1: Feature Categories, Values, and Percentages. *For each of the six feature categories, its possible values (output units) are given. The number given for each value is the percentage of speech segments from the training set that contain that value. The percentages for each column add up to 100%, after accounting for rounding.*

occlusive, PHONATION: voiced, PLACE: coronal, ROUNDNESS: unrounded]. For a given speech vector, the target is to have only one value for each feature category activate, having a value of 1, and all of the rest inactive, them having a value of -1.

Each of the six networks has an input layer of 13 units, three hidden layers, and one output layer. The first hidden layer is the delta layer and has 13 units. It receives connections from the input layer and computes the difference between the current frame and the previous. The second hidden layer is the acceleration layer and also has 13 units. It receives connections from the delta layer and computes the difference between the current delta values and the previous delta values. The third hidden layer is a recurrent layer. It receives connections from the input, delta, and acceleration layers. Also, each hidden unit in the third hidden layer is connected to every other unit within this layer, including itself. Furthermore the connections among the units decay with the distance between the units; that is, the connection between, say, units 1 and 2 will be stronger than the connection between units 1 and 4.

The number of units needed in the recurrent layer for each of the six networks varied. ROUNDNESS, for example, needed as few as 20 hidden units. Adding additional hidden units did not increase its ability to learn while taking away some of these hidden units decreased its ability to learn. CENTRALITY was also at its optimum with 20 hidden units, though I have not checked, due to lack of resources, to see if it could run just as well with fewer hidden units. PHONATION had better results with 40 hidden units than with 20. Finally, PLACE, FRONTBACK, and MANNER needed as many as 80 hidden units. It is possible that these final three networks should be even bigger. To train a network with more than 80 hidden units takes a processing time, and I did not have that amount of time to put into training them.

The rate of decay for the recurrent connections in the third hidden layer can be modified by the spread coefficient (Ström, 1997) My default spread coefficient was 50. I tried decreasing this some in different experimental models. In some networks (CENTRALITY, MANNER, FRONTBACK, PHONATION, and PLACE), varying it decreased the network's ability to learn only slightly. The one exception to this is PLACE, which lost over three percentage points when its spread coefficient was decreased to 20. In ROUNDNESS, using a small spread coefficient (in this case, 5) produced a marginally better network; so, I kept the better value for this network instead of my default.

The output layer receives its connections only from the third hidden layer. The number of units in the output layer is the same as the number of values for each feature. So, there is one unit assigned to each value. For a given frame, the highest activated output unit is the sole winner, and that unit's value is taken to be the value for the feature net. Though, we can also examine the rankings of the activations of the output units; so, if a unit has, say, the second highest activation, we can put it into the list of top two activations. This ordered list is useful when the we want to see an alternative choice to the highest value.

So, there are several components in the architecture to examine a frame's context in time.

NET Num Values	CENTRALITY 4	FRONTBACK 4	MANNER 6	PHONATION 3	PLACE 9	ROUNDNESS 3
Top	85.7	85.4	86.5	93.1	76.4	91.5
Top-2	96.9	97.7	97.0	99.4	90.9	99.8

Table 2: Net Correctness: *For each feature category, the percentage of frames that the respective net correctly classified Also, the percentage where the correct value was one of the top 2 activated units.*

There are the already mentioned overlapping frames, the delta and acceleration layers, and the recurrent connections. Furthermore, there is a time window of +/- 3 frames in the connections that the recurrent layer has with the other layers. The connections will represents three frames before and after the current frame.

Finally, back propagation is used for training.

3 Results

First, I will present the overall correctness results. Then I will discuss the possible weakness of having used cepstral analysis in generating the frames. This will lead into discussions on the imbalance of the training set relates to the learning capacity and on how the nets can learn broader categories of values. Finally, by examining a sample sentence from the test set, I will illustrate how MANNER has learned its values. I will not be giving fine details on all of the nets but, rather, will be pointing out the more interesting aspects from some of the nets.

After being trained on 3548 sentences from the TIMIT speech corpus (Garofolo, 1988), the nets were tested on a separate 1344 sentences from TIMIT. All results given here are from the test set; the test set is the standard testing set used in TIMIT work and does not use any of the speakers from the training set. Using TIMIT for training and testing provided data from a wide variety of American English dialects, both male and female; having this variety provides the nets the opportunity to make good generalisations about spoken American English. Table 2 shows the results from the test runs. The "Top-2" row is of particular value especially with MANNER and PLACE, which both have a wide range of target values, six and nine respectively. While these two nets struggle some in scoring the target value as the highest, they can score it as one of the top two highest values at least 90% of the time.

These high results are based on the effective cepstral analysis done on the speech. While cepstral vectors are a good model of the speech waves, there are certain aspects in speech that it can not model. Nasality and roundness are difficult for cepstral analysis to detect(King, 1998). Table 3 implies the difficulty in determining roundness from the input cepstral vectors. It shows that if the target value is `unrounded`, it correctly identifies the frame as `unrounded` 95.9% of the time. However, if the target is `rounded`, it correctly identifies it as `rounded` 47.2% of the time; most of the time (52.6%) it will misidentify a `rounded` segment as being `unrounded`. It is uncertain how much of the low correctness in identifying `rounded` values is due to the cepstral analysis. This is because the training set was `unrounded` 79% of the time (Table 1). So, much, if not all, of the difference between the `rounded` and `unrounded` correctness can be attributed to there being almost 9 times the number of `unrounded` versus `urounded` samples.

It should be noted here that while ROUNDNESS presents a higher correctness than, say, MANNER, MANNER actually presented a better ability to learn. If both networks were to do their learning by just outputting the most likely value, based on their training data, then ROUNDNESS would classify everything as `unrounded` and MANNER would classify everything as `vowel`. This gives a baseline of 37% for MANNER and 79% for ROUNDNESS. So, manner has managed to improve about 50% from its baseline while roundness has improved only 12% from its own baseline.

The features in the PHONATION net, however, should be represented well by the cepstral coefficients, and it did learn well. Now, the values for this net are not balanced equally in the training

	unrounded	rounded	silence
unrounded	95.9	3.0	1.0
rounded	52.6	47.2	0.2
silence	12.0	0.2	87.9

Table 3: Confusion matrix for ROUNDNESS Net

	voiced	unvoiced	silence
voiced	94.6	4.8	0.5
unvoiced	10.3	88.0	1.7
silence	3.8	5.0	91.3

Table 4: partial Confusion matrix for PHONATION Net

set (Table 1). There are over twice the number of training segments of `voiced` than there are of `unvoiced`. As a result, the `voiced` was learned the best as it has the most training data. Furthermore, it has learned in such a way that if it misclassifies `unvoiced`, it does so by giving a `voiced` classification (though this could be a result of the large number of `voiced` samples.

In the PLACE net, there is some generalisation of broader classes. Table 5 gives a partial confusion matrix. This net has only mediocre performance for the vowel related values of `low` and `medium`, but `high` has a more acceptable performance. However, with `low` and `medium`, even if the net incorrectly identifies these values, it is likely to label the segment as one of the other two vowel values. So, it is not marking the frame in a totally unrelated way. Rather, even if the net does not make the right guess when dealing with one of these values, it still knows what general type of frame it is. That is, it knows it is looking at something with a vowel like place: `high`, `medium`, or `low`.

Finally, Figure 1 illustrates how a net responds to a test sentence. Notice the following in it. If often does a good jobs, particularly with `vowel`, of giving the favoured unit a very high score (+1). Likewise, it often keeps the non-targeted values a very low score (-1). Regarding the VOWEL unit, the net gives `vowel` frames high scores consistently. It does something interesting with the `approximant` /er/. That is, the frame was marked that /er/ is only an `approximant`; however, the net divided that phone in half, saying that the first part was a `vowel` while the second part was an `approximant`. This is consistent with the pronunciation of the "ere" in "were"; so the net has tried to overcome a misleading label. The FRICATIVE unit has mixed results. It gets the /z/ phone correct, but has trouble with the /dh/ phone and has some small rises in the graph where it should be at -1. The APPROXIMANT unit does identify the first /w/ but not with great conviction. It does identify the final /w/ and the following /er/, with the exception noted with the VOWEL unit previously. The NASAL unit does a good job in identifying the single `nasal`, the /n/, considering, as mentioned earlier, that the cepstral coefficients are not a good model for nasals. The NASAL unit does stay near -1 when it is supposed to with the notable exception that it will rise a little when it encounters most other consonants. The OCCLUSIVE identifies the sole `occlusive`, /dx/ correctly but has some unexplained bumps in the other parts of its graph. Finally, the SILENCE unit is very accurate.

	low	medium	high	OTHER
low	48.7	34.6	8.2	8.5
medium	14.1	52.6	15.3	18.0
high	2.0	8.8	78.5	10.8

Table 5: partial Confusion matrix for PLACE Net

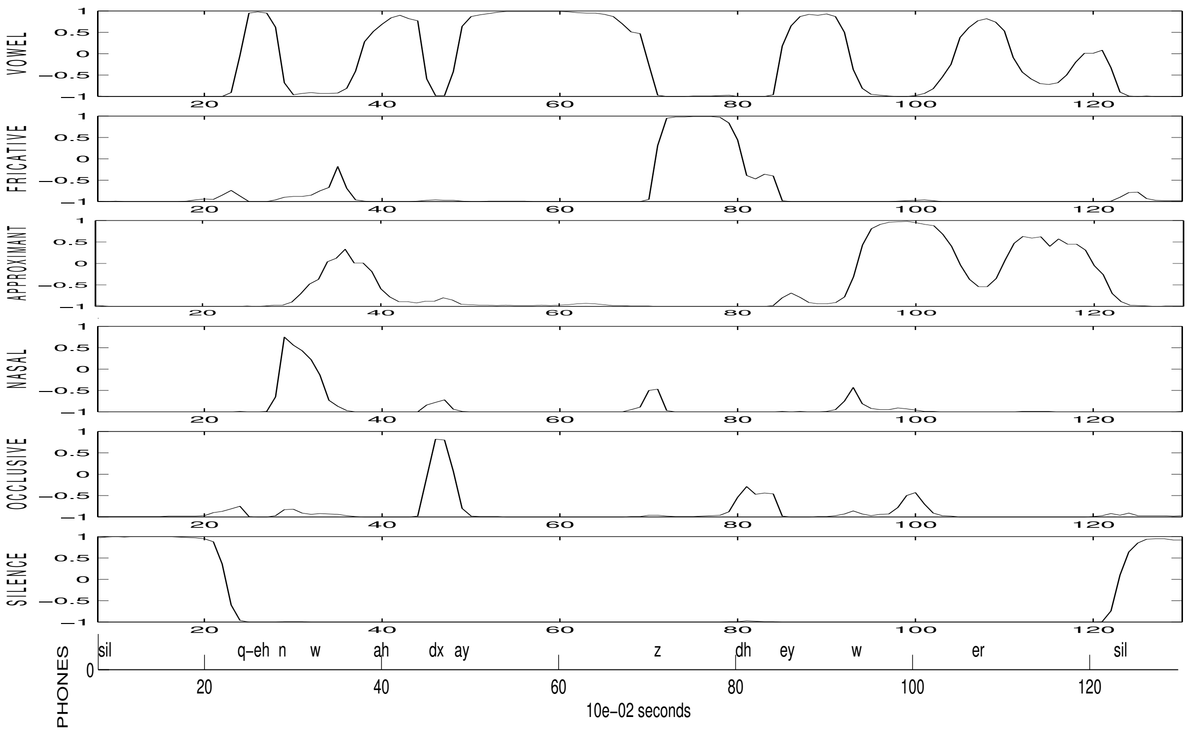

Figure 1: MANNER output for phrase: *"And what eyes they were." It is plotted from 80 to 1280 ms (milliseconds). For each frame the activation of each of the six output units is plotted. The range of each plot is from -1 (inactive) to 1 (active). (In ideal learning, for each plot in time, one output unit would be 1 and the rest would be -1). The phone transcriptions are TIMIT-style; the 'q-eh' is really the separate phones 'q' and 'eh' that were written together because of space constraints. The overall correctness for this net on this sentence is 87.0%. Since this is around the average correctness for this net (Table 2), this is typical output for this net.*

4 Discussion

These results have shown the ability of neural networks to learn phonetic features in speech. While they are not always correct, at the same time they show their ability to generalise what they have seen about the data and even to learn the relations between different values for a feature.

They learned phonetic features with the best correctness when the values are kept few and simple. The nets with only three possible values, PHONATION and ROUNDNESS learned better than those with four or six values. The net with the largest number of values, PLACE at nine, had the lowest accuracy of them all. However, it is with these larger nets that they show their ability to generalise among related values.

Now even when they are inaccurate, they are sometimes just labelling the frame with a related part of speech (such as labelling a `medium` as a `low`). So, we can still gather some useful information from the net when it has given the wrong answer. Furthermore, from Table 2, if the correct output is not the highest output unit, it is very likely that it has received the second best score.

A better balanced training set would be ideal. This would enable all the possible outputs of a neural net to have an equal chance to learn. It would also encourage the back propagation algorithm to better learn the values from these nets that were poorly learned.

Finally, further tests are being done to see how building a regular phonemic speech recogniser from these net outputs compares to building the recogniser from the original cepstral coefficients. Additionally, this neural network project is being used in ongoing research into doing speech recognition based on syllable recognition.

5 Acknowledgements

I would like to thank Mark Ellison for his supervision in training the networks and Simon King for suggesting the topic for this project and for providing valuable insight into the project's results.

References

Garofolo, J. S. (1988). *Getting started with the DARPA TIMIT CD-ROM: An acoustic phonetic continuous speech database*. National Institute of Standards and Technology (NIST), Gaithersburgh, MD.

King, S. A. (1998). Personal communication.

Kirchhoff, K. (1996). Syllable-level desynchronisation of phonetic features for speech recognition. In *International Conference on Spoken Language Processing*, volume 4.

Owens, F. J. (1993). *Signal Processing of Speech*. The Macmillan Press Ltd.

Ström, N. (1997). Phoneme probability estimation with dynamic sparsely connected artificial neural networks. *The Free Speech Journal*.

Young, S., Jansen, J., Odell, J., Ollason, D., and Woodland, P. (1996). *HTK manual*. Entropic.