Chapter 5

Experiments

This chapter reports the results obtained on phoneme segmentation and recognition experiments. Section 5.1 describes the segmentation of speech into phonemes, where Section 5.1.1 gives results for HMM based segmentation, and Section 5.1.2 for BRNN based segmentation. Section 5.2 discusses results obtained for the baseline phoneme recognition experiments. Finally, Section 5.3 gives results on the recognition of speech using segmentation information. Please note that in this chapter, the word "optimised" is used to refer to empirical optimisation, and is thus not strictly correct in a mathematical sense.

5.1 Experiment 1: Speech segmentation

The segmentation of speech into phonemes, as discussed in detail in Chapter 3, can be done in two ways. The first is the use of hidden Markov models, where the Viterbi algorithm is used to find the most likely phone sequence, given the speech utterance. Since the times between the model transitions are known, these boundary locations can be used as the segmentation of the speech signal. The second segmentation method, is to use a bi-directional recurrent neural network, that estimates the probability of a

boundary and no boundary, given the speech vectors. A postprocessor then determines the actual boundary locations, through the use of a threshold function. Both of these techniques will be considered here.

5.1.1 Hidden Markov models

For the segmentation of speech, using hidden Markov models, four factors that have an influence on the segmentation accuracy, are investigated. The first is the number of mixtures used in the output probability distribution of each HMM. The second is the window size, or allowed difference between the true and estimated phoneme boundary positions. The effects of both a language model and embedded re-estimation, are also investigated.

Number of HMM mixtures

Purpose: The purpose of this experiment is to determine the effect that the number of mixtures in the HMM output probability distribution, has on the segmentation accuracy of an HMM based segmentation system.

Experimental setup: Speech utterances are recognised using trained HMMs, and the recognition results converted into a set of phoneme boundary locations, as described in Chapter 3. The HMMs are trained on 3696 files of the TIMIT training set, and evaluated by segmenting the 1344 files of the TIMIT test set (the full test set). No language model is used, and embedded re-estimation is not performed. The window size is fixed at 3 (20 ms).

Results: Figure 5.1 shows the effect of the number of mixtures in the HMM output probability density has on the segmentation accuracy. The details of these results are given in Table 5.1. The table shows the average accuracy and percentage of boundaries correctly identified, the number of boundaries correctly matched (hits), insertions and

deletions, over the full test set. It can be seen that the number of mixtures does not have a significant influence on the segmentation accuracy. The best segmentation accuracy, at a window size of 3, is obtained with 8 mixtures, and is equal to 71.23%, with a percentage correct of 87.07%. The average number of hits, insertions and deletions, per utterance (sentence) over the complete test data set, is equal to 31.40, 5.40, and 4.64, respectively. The average number of true boundaries per utterance is 36.05 and the average number of observations (frames) per utterance is 305.71. It should be noted that even though the best performance was obtained with 8 mixtures, 2 mixtures would probably be sufficient to model the acoustic space due to the fact that the TIMIT database consists of male and female speakers.

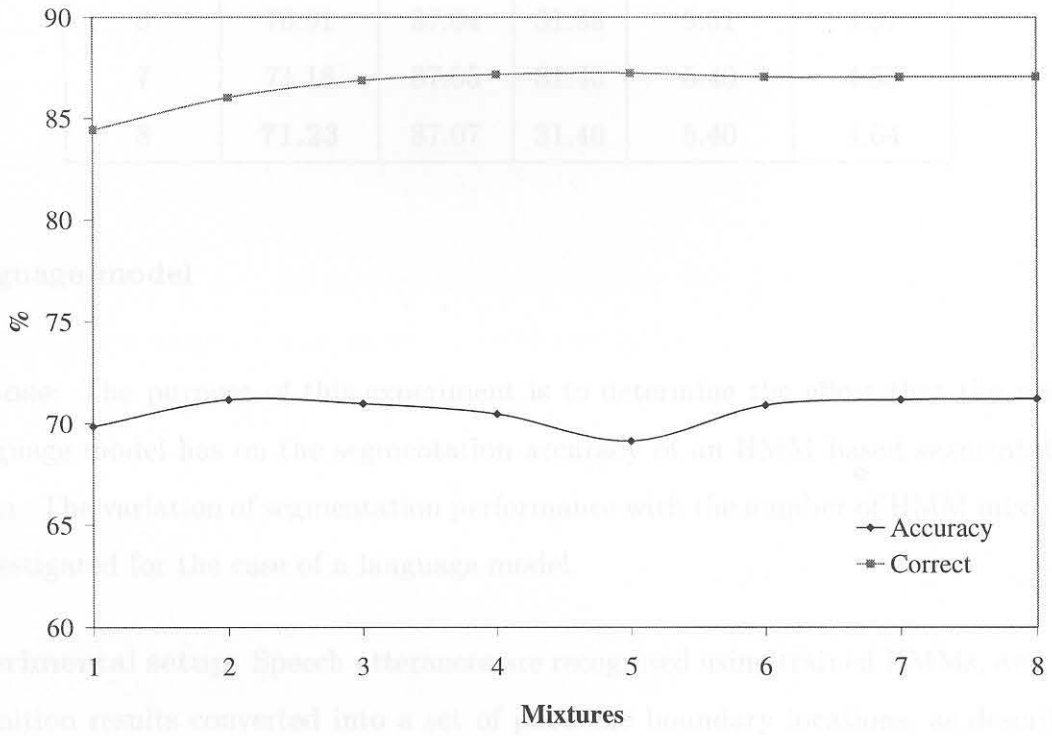

Figure 5.1: Effect of the number of mixtures in the HMMs, on the segmentation performance. No language model is used, and no embedded re-estimation performed.

Table 5.1: Numerical results of the segmentation performance obtained with the HMM based segmentation system, for different numbers of mixtures in the output probability densities. No language model or embedded re-estimation is performed.

Mixtures	Accuracy (%)	Correct (%)	Hits (#)	Insertions (#)	Deletions (#)
1	69.86	84.44	30.40	**5.04**	5.64
2	71.18	86.03	30.97	5.12	5.07
3	70.98	86.87	31.28	5.45	4.77
4	70.47	87.17	31.41	5.71	4.64
5	69.13	**87.23**	**31.45**	6.20	**4.60**
6	70.91	87.04	31.38	5.51	4.67
7	71.18	87.05	31.40	5.40	4.65
8	**71.23**	87.07	31.40	5.40	4.64

Language model

Purpose: The purpose of this experiment is to determine the effect that the use of a language model has on the segmentation accuracy of an HMM based segmentation system. The variation of segmentation performance with the number of HMM mixtures is investigated for the case of a language model.

Experimental setup: Speech utterances are recognised using trained HMMs, and the recognition results converted into a set of phoneme boundary locations, as described in Chapter 3. The HMMs are trained on 3696 files of the TIMIT training set, and evaluated by segmenting the 1344 files of the TIMIT test set (the full test set). A bigram language model, calculated on the text data of the training set, is used and embedded re-estimation is not performed. The window size is fixed at 3 (20 ms).

Results: Table 5.2 shows the variation of the segmentation performance with the number of mixtures in the HMM output probability density, when a language model

Table 5.2: Summary of results showing the segmentation performance obtained with the HMM based segmentation system, for different numbers of mixtures in the output probability densities. A bigram language model is used and no embedded re-estimation is performed.

Mixtures	Accuracy (%)	Correct (%)	Hits (#)	Insertions (#)	Deletions (#)
1	72.85	80.56	29.04	**2.63**	7.02
2	75.08	82.84	29.85	2.66	6.20
3	75.36	83.53	30.12	2.76	5.93
4	75.37	83.95	30.27	2.90	5.77
5	74.95	84.02	30.31	3.07	5.73
6	75.58	84.10	30.35	2.88	5.69
7	**75.69**	**84.11**	**30.35**	2.85	**5.69**
8	75.62	84.03	30.33	2.84	5.72

is used. The table shows the average accuracy and percentage of boundaries correctly identified, the number of boundaries correctly matched (hits), insertions and deletions, over the full test set. It can be seen that number of mixtures does not have a significant influence on the segmentation accuracy. The language model has the additional effect of smoothing the segmentation performance, as the number of insertions is decreased. The number of hits, however, decreases slightly, and the number of deletions slightly increases. This explains the slight decrease in the percentage correct, over the case when no language model is used. The best segmentation accuracy is obtained with 7 mixtures, and is equal to 75.69%, with a percentage correct of 84.11%. The average number of hits, insertions and deletions is equal to 30.35, 2.85, and 5.69, respectively. The average number of true boundaries is 36.05 and the average number of observations (frames) is 305.71. Here 3 mixtures would probably be sufficient for the segmentation task, even though the best performance was obtained with 7 mixtures.

Embedded re-estimation

Purpose: The purpose of this experiment is to determine the effect that the use of embedded re-estimation has on the segmentation accuracy of an HMM based segmentation system. The variation of segmentation performance with the number of HMM mixtures is investigated both when no language model is used, and when a language model is used.

Experimental setup: Speech utterances are recognised using trained HMMs, and the recognition results converted into a set of phoneme boundary locations, as described in Chapter 3. The HMMs are trained on 3696 files of the TIMIT training set, and evaluated by segmenting the 1344 files of the TIMIT test set (the full test set). When a language model is used, a bigram language model is chosen. Embedded re-estimation is performed. The window size is fixed at 3 (20 ms).

Results: Tables 5.3 and 5.4 show the variation of the segmentation performance with the number of mixtures in the HMM output probability distribution, when both a language model and no language model is used, and embedded re-estimation is performed. The tables shows the average accuracy and percentage of boundaries correctly identified, the number of boundaries correctly matched (hits), insertions and deletions, over the full test set. It can be seen that number of mixtures does not have a significant influence on the segmentation accuracy. The language model has the additional effect of smoothing the segmentation performance, as the number of insertions is decreased. The number of hits, however, decreases slightly, and the number of deletions slightly increases. This explains the slight decrease in the percentage correct, when no language model is used. When a language model is not used, the best segmentation accuracy is obtained with 7 mixtures, and is equal to 70.27%, with a percentage correct of 88.04%. The average number of hits, insertions and deletions is equal to 31.76, 6.07, and 4.29, respectively. When a language model is used, the best segmentation accuracy is obtained with 4 mixtures, and is equal to 75.47%, with a percentage correct of 85.28%. The average number of hits, insertions and deletions is equal to 30.77, 3.31, and 5.27,

Table 5.3: Numerical results of the segmentation performance obtained with the HMM based segmentation system, for different numbers of mixtures in the output probability densities. No language model is used and embedded re-estimation is performed.

Mixtures	Accuracy (%)	Correct (%)	Hits (#)	Insertions (#)	Deletions (#)
1	69.00	85.83	30.90	**5.82**	5.15
2	68.28	87.18	31.42	6.49	4.63
3	69.47	87.90	31.68	6.34	4.37
4	69.52	88.29	31.83	6.42	4.22
5	68.14	**88.47**	**31.91**	6.97	**4.14**
6	69.73	88.03	31.76	6.25	4.29
7	**70.27**	88.04	31.76	6.07	4.29
8	69.70	87.43	31.53	6.07	4.51

respectively. The average number of true boundaries is 36.05 and the average number of observations (frames) is 305.71. Embedded re-estimation thus results in a slight decrease in the segmentation performance [71.23% vs 70.27% (no LM) and 75.69 vs 75.47 (with LM)].

Window size

Purpose: The purpose of this experiment is to determine the effect that the window size has on the segmentation accuracy of an HMM based segmentation system. The variation of segmentation performance with the window size is investigated both when no language model is used, and when a language model is used.

Experimental setup: Speech utterances are recognised using trained HMMs, and the recognition results converted into a set of phoneme boundary locations, as described in Chapter 3. The HMMs are trained on 3696 files of the TIMIT training set, and

Table 5.4: Numerical results of the segmentation performance obtained with the HMM based segmentation system, for different numbers of mixtures in the output probability densities. A bigram language model is used and embedded re-estimation is performed.

Mixtures	Accuracy (%)	Correct (%)	Hits (#)	Insertions (#)	Deletions (#)
1	73.20	82.49	29.72	**3.18**	6.32
2	73.78	84.14	30.35	3.52	5.70
3	74.92	84.88	30.62	3.40	5.43
4	**75.47**	85.28	30.77	3.31	5.27
5	75.03	**85.34**	**30.80**	3.51	**5.24**
6	75.23	85.24	30.79	3.40	5.26
7	75.40	85.22	30.78	3.32	5.26
8	74.34	84.83	30.64	3.56	5.41

evaluated by segmenting the 1344 files of the TIMIT test set (the full test set). When a language model is used, a bigram language model is chosen. Embedded re-estimation is performed. The number of mixtures is fixed at 8.

Results: Figure 5.2 shows the variation of the segmentation performance with the window size, when no language model is used, and embedded re-estimation is performed. These results are given numerically in Table 5.5. Table 5.6 shows the results when a language model is used. The tables shows the average accuracy and percentage of boundaries correctly identified, the number of boundaries correctly matched (hits), insertions and deletions, over the full test set. It can be seen that the window size has a significant effect on the perceived segmentation performance. The use of a language model increases the segmentation performance, as the number of insertions is decreased. Many researchers mention segmentation accuracy at a window size of 20 ms, or a window size of 3, as used in this dissertation. When a language model is not used, the segmentation performance at a window size of 3 is equal to 69.70%, with a percentage correct of 87.43%. The average number of hits, insertions and deletions is

equal to 31.53, 6.07, and 4.51, respectively. When a language model is used, the best segmentation accuracy at a window size of 3, is equal to 74.34%, with a percentage correct of 84.83%. The average number of hits, insertions and deletions is equal to 30.64, 3.56, and 5.41, respectively. The average number of true boundaries is 36.05 and the average number of observations (frames) is 305.71.

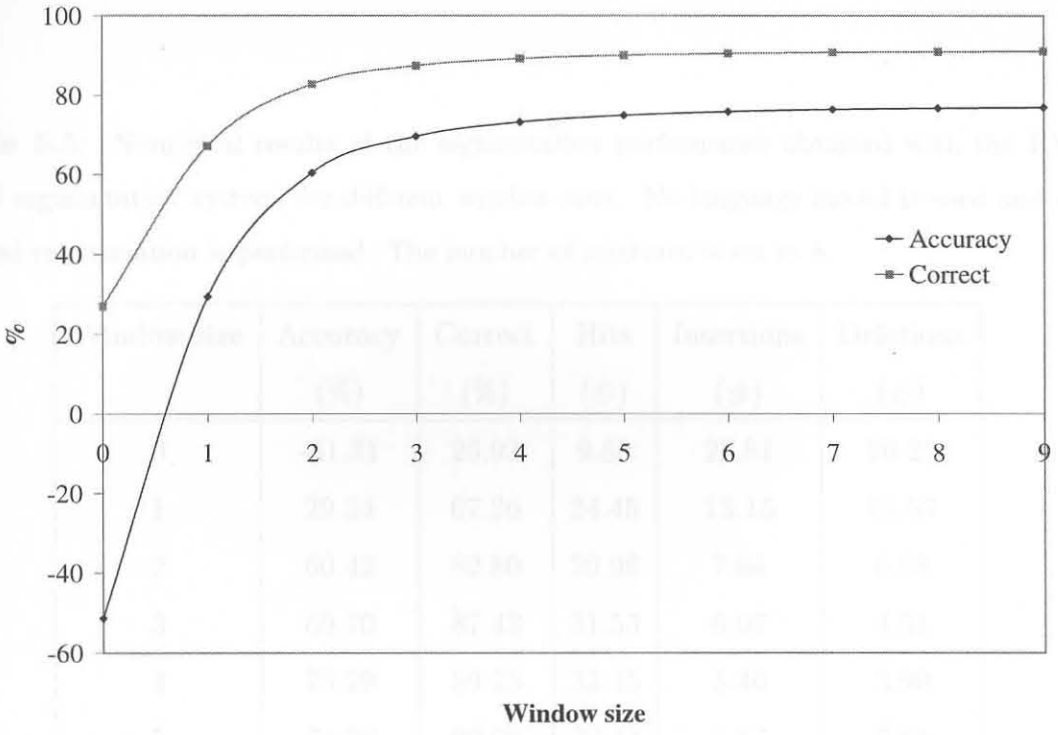

Figure 5.2: Effect of the window size (see Section 3.4 for a discussion of the window size and its mapping to time) on the segmentation performance. No language model is used, and embedded re-estimation performed. The number of mixtures is equal to 8.

The results of Section 5.1.1 (segmentation using HMMs) are summarised and compared in Chapter 6.

5.1.2 Recurrent neural network

For the segmentation of speech, using a recurrent neural network, or more precisely, a bi-directional recurrent neural network, three main parameters have an influence on the

Table 5.5: Numerical results of the segmentation performance obtained with the HMM based segmentation system, for different window sizes. No language model is used and embedded re-estimation is performed. The number of mixtures is set to 8.

Window size	Accuracy (%)	Correct (%)	Hits (#)	Insertions (#)	Deletions (#)
0	-51.31	26.93	9.80	27.81	26.25
1	29.34	67.26	24.45	13.15	11.59
2	60.42	82.80	29.96	7.64	6.08
3	69.70	87.43	31.53	6.07	4.51
4	73.29	89.23	32.15	5.46	3.90
5	74.98	90.08	32.43	5.17	3.61
6	75.88	90.53	32.59	5.02	3.46
7	76.45	90.81	32.68	4.92	3.37
8	76.72	90.95	32.73	4.88	3.32
9	76.93	91.05	32.76	4.84	3.28

Table 5.6: Numerical results of the segmentation performance obtained with the HMM based segmentation system, for different window sizes. A bigram language model is used and embedded re-estimation is performed. The number of mixtures is set to 8.

Window size	Accuracy (%)	Correct (%)	Hits (#)	Insertions (#)	Deletions (#)
0	-41.37	26.97	9.85	24.34	26.19
1	36.23	65.77	23.96	10.25	12.10
2	65.34	80.32	29.12	5.08	6.93
3	74.34	84.83	30.64	3.56	5.41
4	78.10	86.71	31.27	2.93	4.78
5	79.83	87.57	31.55	2.64	4.49
6	80.74	88.03	31.71	2.49	4.33
7	81.31	88.31	31.80	2.39	4.24
8	81.65	88.48	31.86	2.34	4.19
9	81.84	88.57	31.89	2.31	4.16

segmentation accuracy. The first is the number of hidden nodes in the network, that affects the network modeling capability. The second is the value of the threshold in the postprocessor, used to decide whether the probability of a boundary, as estimated by the neural network, is high enough to be regarded as a boundary or not. The final parameter that has a major influence on segmentation accuracy, is the window size, or allowed difference between the true and estimated phoneme boundary positions.

Number of hidden nodes

Purpose: The purpose of this experiment is to determine the effect that the number of hidden nodes in the bi-directional recurrent neural network has on the segmentation accuracy of a BRNN based segmentation system.

Experimental setup: The bi-directional recurrent neural network is trained using the 3696 TIMIT training files. The 1344 TIMIT test files (full test set) are then segmented. The neural network is trained for 250 iterations of backpropagation through time, with a learning rate of 0.001, with the weights in the network initialised uniformly between -0.15 and 0.15. For each number of hidden nodes, the network chosen is the one at the iteration which gives the best performance on the test set. The number of forward hidden nodes is equal to the number of hidden nodes, and the number backward hidden nodes as well. The total number of hidden nodes is thus equal to twice the number of hidden nodes, as mentioned here. In this experiment, the boundary threshold is fixed at 0.35 and the window size at 3.

Results: Figure 5.3 shows the effect that the number of hidden nodes has on the segmentation accuracy. These results are summarised numerically in Table 5.7. The table shows the average accuracy and percentage of boundaries correctly identified, the number of boundaries correctly matched (hits), insertions and deletions, over the full test set. It can be seen that number of hidden nodes has a significant effect on the segmentation performance. The best segmentation accuracy is obtained with 60

hidden nodes, and is equal to 80.12%, with a percentage correct of 86.20%. The average number of hits, insertions and deletions is equal to 31.14, 2.14, and 4.91, respectively. The average number of true boundaries is 36.05 and the average number of observations (frames) is 305.71.

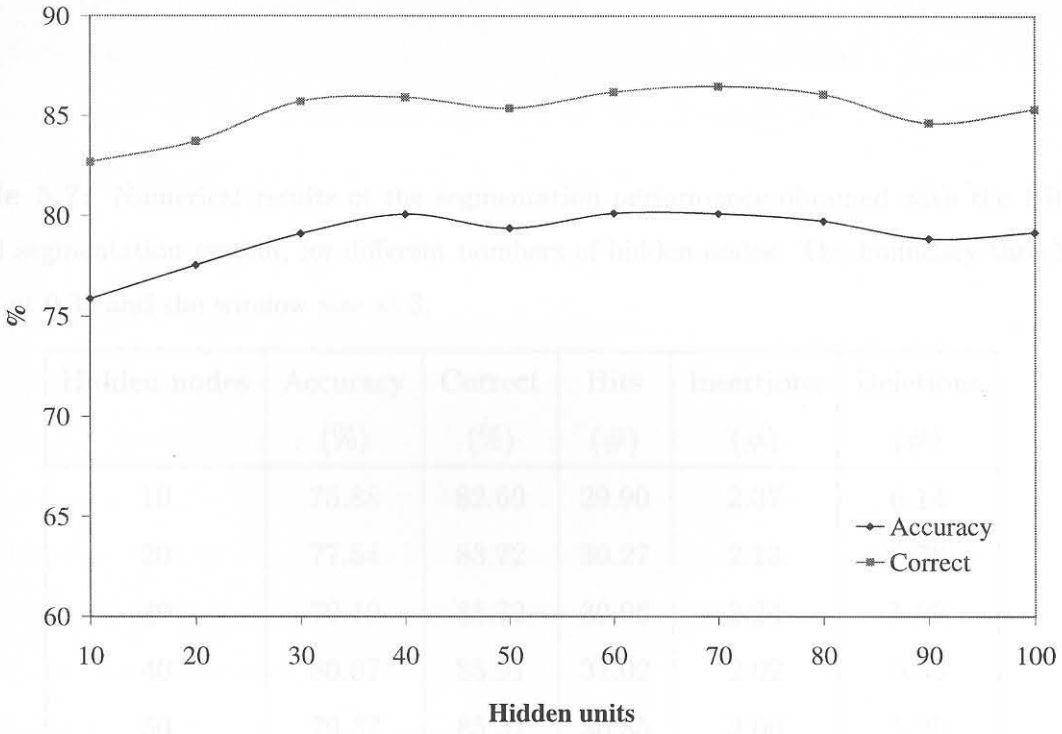

Figure 5.3: Effect of the number of neural network hidden nodes on the segmentation performance. The boundary threshold is set at 0.35 and the window size at 3.

Boundary threshold

Purpose: The purpose of this experiment is to determine the effect that the boundary threshold in the postprocessor of the neural network, has on the segmentation accuracy of a BRNN based segmentation system.

Experimental setup: The bi-directional recurrent neural network is trained using the 3696 TIMIT training files. The 1344 TIMIT test files (full test set) are then segmented. The neural network is trained for 250 iterations of backpropagation through time, with

Table 5.7: Numerical results of the segmentation performance obtained with the BRNN based segmentation system, for different numbers of hidden nodes. The boundary threshold is set at 0.35 and the window size at 3.

Hidden nodes	Accuracy (%)	Correct (%)	Hits (#)	Insertions (#)	Deletions (#)
10	75.88	82.69	29.90	2.37	6.14
20	77.54	83.72	30.27	2.13	5.78
30	79.10	85.72	30.96	2.24	5.09
40	80.07	85.91	31.02	2.02	5.03
50	79.37	85.37	30.85	2.06	5.20
60	**80.12**	86.20	31.14	2.14	4.91
70	80.10	**86.48**	**31.22**	2.20	**4.82**
80	79.73	86.08	31.07	2.23	4.98
90	78.86	84.65	30.58	**1.99**	5.46
100	79.18	85.34	30.84	2.12	5.21

a learning rate of 0.001, with the weights in the network initialised uniformly between -0.15 and 0.15. For each number of hidden nodes, the network chosen is the one at the iteration (between 1 and 250) which gives the best performance on the test set. The number of forward hidden nodes (N_f) is equal to the number of backward hidden nodes (N_b). We call this the number of hidden nodes ($N_f = N_b = N_h$). The total number of hidden nodes (N_{htot}) is thus equal to twice the number of hidden nodes (N_h), as mentioned here. In this experiment $N_f = N_b = N_h$ is fixed at 50. The window size is set to 1 (the true and estimated boundaries may only differ by 1 frame).

Results: Figure 5.4 shows the effect that the boundary threshold has on the segmentation accuracy. These results are given numerically in Table 5.8. The table shows the average accuracy and percentage of boundaries correctly identified, the number of boundaries correctly matched (hits), insertions and deletions, over the full test set. It can be seen that the boundary threshold has a significant effect on the segmentation performance. The best segmentation accuracy is obtained with a boundary threshold of 0.35, and is equal to 68.41%, with a percentage correct of 79.89%. The average number of hits, insertions and deletions is equal to 28.95, 3.95, and 7.10, respectively. The average number of true boundaries is 36.05 and the average number of observations (frames) is 305.71.

Window size

Purpose: The purpose of this experiment is to determine the effect that the window size has on the segmentation accuracy of a BRNN based segmentation system.

Experimental setup: The bi-directional recurrent neural network is trained using the 3696 TIMIT training files. The 1344 TIMIT test files (full test set) are then segmented. The neural network is trained for 250 iterations of backpropagation through time, with a learning rate of 0.001, with the weights in the network initialised uniformly between -0.15 and 0.15. For each number of hidden nodes, the network chosen is the one at

Table 5.8: Numerical results of the segmentation performance obtained with the BRNN based segmentation system, for different boundary threshold values. The number of hidden units is set at 50 (for both the forward and backward hidden nodes) and the window size at 1.

Threshold	Accuracy (%)	Correct (%)	Hits (#)	Insertions (#)	Deletions (#)
0.00	25.93	**86.53**	**31.28**	20.42	**4.77**
0.05	56.83	85.81	31.02	9.93	5.03
0.10	61.84	85.13	30.78	7.99	5.27
0.15	64.94	84.38	30.51	6.69	5.53
0.20	66.60	83.44	30.18	5.78	5.86
0.25	67.74	82.52	29.85	5.07	6.19
0.30	68.20	81.31	29.44	4.51	6.61
0.35	**68.42**	79.89	28.95	3.96	7.10
0.40	68.25	79.09	28.66	3.76	7.38
0.45	67.85	77.32	28.05	3.29	8.00
0.50	66.89	75.05	27.24	2.84	8.81
0.55	65.42	72.55	26.39	2.48	9.66
0.60	63.25	69.20	25.22	2.07	10.83
0.65	59.67	64.48	23.54	1.68	12.50
0.70	54.13	57.94	21.20	1.34	14.85
0.75	45.15	48.03	17.60	1.01	18.45
0.80	33.00	35.23	12.90	0.77	23.15
0.85	18.16	19.98	7.32	0.63	28.73
0.90	5.49	7.16	2.60	0.58	33.44
0.95	-0.46	0.77	0.27	0.42	35.77
1.00	-0.60	0.63	0.22	**0.42**	35.82

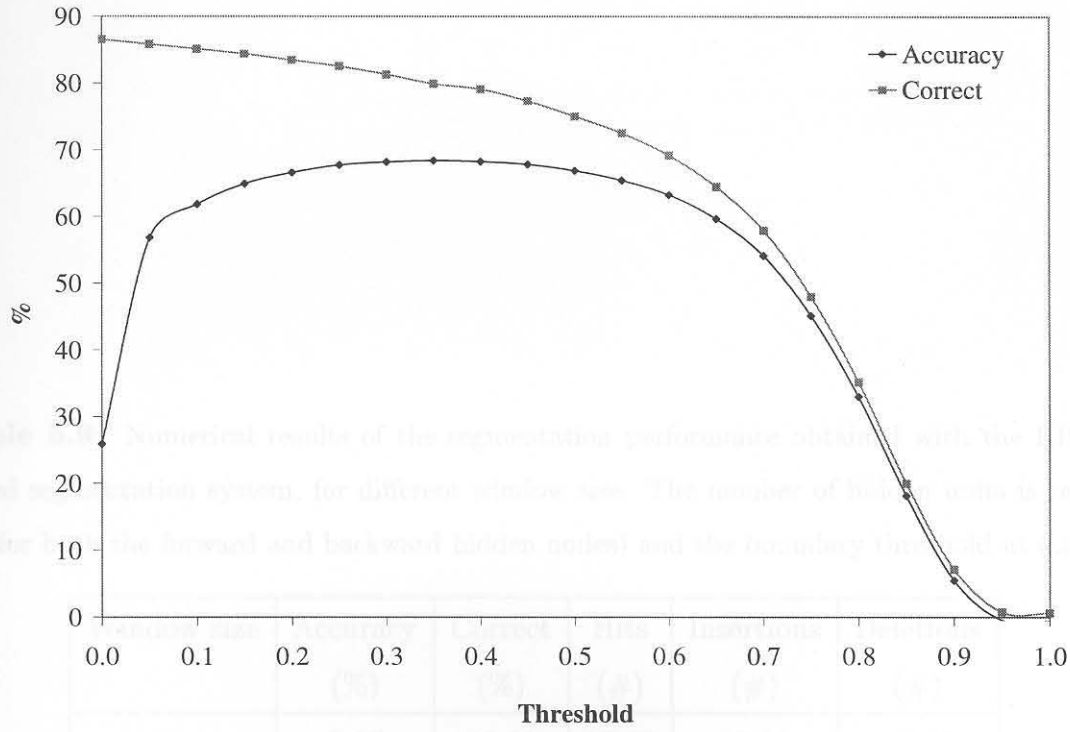

Figure 5.4: Effect of the threshold used at the neural network outputs to decide in favour of a boundary or not. The number of hidden nodes is 50 and the window size is 1.

the iteration (1 to 250) which gives the best performance on the test set. In this experiment, $N_f = N_b = N_h$ is fixed at 50. The boundary threshold is set at 0.35.

Results: Figure 5.5 shows the effect that the window size has on the segmentation accuracy. These results are given numerically in Table 5.9. The table shows the average accuracy and percentage of boundaries correctly identified, the number of boundaries correctly matched (hits), insertions and deletions, over the full test set. It can be seen that the window size has a significant effect on the segmentation performance. The accuracy at a window size of 20 ms, or a window size of 3, as used in this dissertation, is equal to 79.37%, with a percentage correct of 85.37%. The average number of hits, insertions and deletions is equal to 30.85, 2.06, and 5.20, respectively. The average number of true boundaries is 36.05 and the average number of observations (frames) is 305.71.

Table 5.9: Numerical results of the segmentation performance obtained with the BRNN based segmentation system, for different window size. The number of hidden units is set at 50 (for both the forward and backward hidden nodes) and the boundary threshold at 0.35.

Window size	Accuracy (%)	Correct (%)	Hits (#)	Insertions (#)	Deletions (#)
0	5.27	48.31	17.56	15.35	18.49
1	68.42	79.89	28.95	3.96	7.10
2	76.66	84.01	30.39	2.52	5.66
3	79.37	85.37	30.85	2.06	5.20
4	80.36	85.86	31.02	1.89	5.03
5	80.86	86.11	31.10	1.81	4.94
6	81.14	86.25	31.15	1.76	4.90
7	81.34	86.35	31.18	1.72	4.86
8	81.47	86.41	31.20	1.70	4.84
9	81.56	86.46	31.22	1.70	4.83

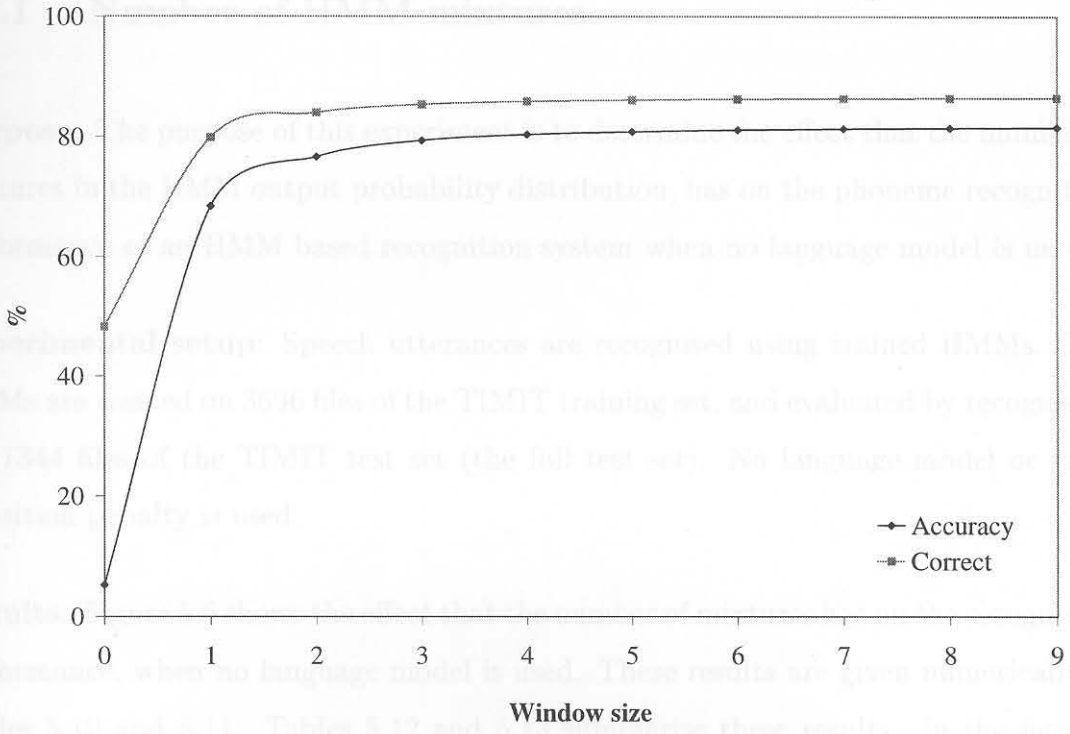

Figure 5.5: Effect of the window size on the segmentation performance of the BRNN. The number of hidden units is 50 and the boundary threshold is set at 0.35.

The results of Section 5.1.2 (segmentation using HMMs) are summarised and compared in Chapter 6. In conclusion it can be seen that the BRNN segments speech better than the HMM-based approach for TIMIT data not containing any noise.

5.2 Experiment 2: Speech recognition (baseline)

This section reports on the phoneme recognition performance of the baseline phoneme recogniser, based on HTK, as discussed in Chapter 4. Four different experiments are conducted. The first is to evaluate the effect of the number of mixtures in the HMM output probability distribution. The second and third experiments investigate the effect of a language model and word transition penalty, respectively. Finally, the combined use of both a language model and word transition penalty is investigated.

5.2.1 Number of HMM mixtures

Purpose: The purpose of this experiment is to determine the effect that the number of mixtures in the HMM output probability distribution, has on the phoneme recognition performance of an HMM based recognition system when no language model is used.

Experimental setup: Speech utterances are recognised using trained HMMs. The HMMs are trained on 3696 files of the TIMIT training set, and evaluated by recognising the 1344 files of the TIMIT test set (the full test set). No language model or word transition penalty is used.

Results: Figure 5.6 shows the effect that the number of mixtures has on the recognition performance, when no language model is used. These results are given numerically in Tables 5.10 and 5.11. Tables 5.12 and 5.13 summarise these results. In the figures, "HERest" indicates the use of additional embedded re-estimation iterations (see [3] for further details). When a language model and embedded re-estimation are not used, the maximum percentage correct is 62.17% and accuracy 52.37%, with 31554 hits, 4915 deletions, 14285 substitutions, and 4975 insertions. When no language model is used, but embedded re-estimation is performed, the maximum percentage correct is 64.80% and accuracy 54.08%, with 32890 hits, 4308 deletions, 13556 substitutions, and 5440 insertions. There are a total of 50754 phonemes in the complete test set.

5.2.2 Language model

Purpose: The purpose of this experiment is to determine the effect that the number of mixtures in the HMM output probability distribution, has on the phoneme recognition performance of an HMM based recognition system when a language model is used.

Experimental setup: Speech utterances are recognised using trained HMMs. The HMMs are trained on 3696 files of the TIMIT training set, and evaluated by recognising

Table 5.10: Numerical results of the baseline HMM recognition performance variation with the number of mixtures, when no language model is used, and no embedded re-estimation is performed.

Mixtures	Correct (%)	Accuracy (%)	Hits (#)	Deletions (#)	Substitutions (#)	Insertions (#)
1	53.37	44.64	27087	6204	17463	**4428**
2	57.52	48.32	29192	5570	15992	4670
3	59.26	49.22	30079	5145	15530	5096
4	60.73	49.96	30823	4983	14948	5468
5	61.56	49.66	31244	**4852**	14658	6041
6	62.04	51.91	31487	4976	14291	5141
7	61.97	52.08	31452	4966	14336	5017
8	**62.17**	**52.37**	**31554**	4915	**14285**	4975

Table 5.11: Numerical results of the baseline HMM recognition performance variation with the number of mixtures, when no language model is used, and embedded re-estimation is performed.

Mixtures	Correct (%)	Accuracy (%)	Hits (#)	Deletions (#)	Substitutions (#)	Insertions (#)
1	55.87	45.54	28354	5304	17096	**5243**
2	60.12	48.02	30514	4602	15638	6142
3	62.41	50.52	31674	4339	14741	6032
4	62.98	50.80	31963	4181	14610	6179
5	63.43	49.76	32194	**4097**	14463	6937
6	63.97	52.45	32469	4176	14109	5846
7	64.06	52.94	32515	4218	14021	5644
8	**64.80**	**54.08**	**32890**	4308	**13556**	5440

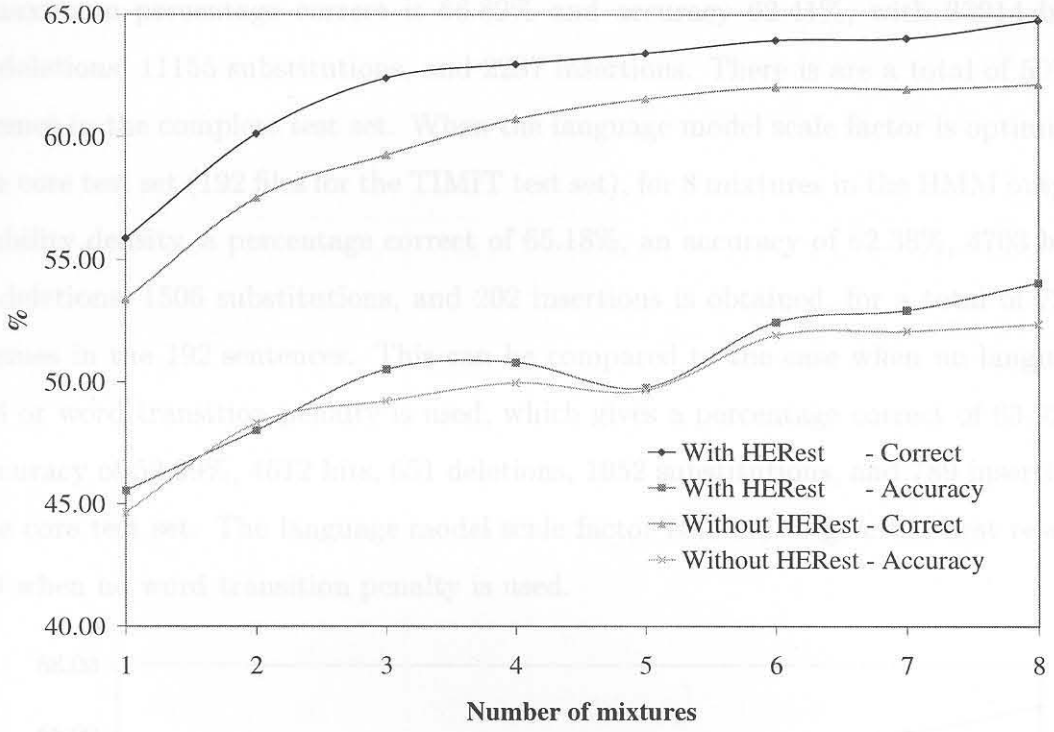

Figure 5.6: Effect of the number of mixtures on HMM phoneme recognition performance. No language model is used.

the 1344 files of the TIMIT test set (the full test set). A bigram language model is used. In the figures given in this experiment, the language model scale factor is set to 2.0. In order to determine the "optimal" value of the language model scale factor (s in Equation (2.45)), it is varied from 0.0 to 10.0 in steps of 1.0. The language model scale factor that results in the highest recognition accuracy is seen as the "optimal" value. A word transition penalty is not used.

Results: Figure 5.7 shows the effect that the number of mixtures has on the recognition performance, when a language model is used. These results are given numerically in Tables 5.12 and 5.13. In the figures, "HERest" indicates the use of additional embedded re-estimation iterations (see [3] for further details). When a language model is used, and embedded re-estimation is not used, the maximum percentage correct is 64.21% and accuracy 60.54%, with 32589 hits, 6695 deletions, 11470 substitutions, and 1861 insertions. When a language model is used, but embedded re-estimation is performed,

the maximum percentage correct is 66.82% and accuracy 62.41%, with 33914 hits, 5685 deletions, 11155 substitutions, and 2237 insertions. There is are a total of 50754 phonemes in the complete test set. When the language model scale factor is optimised on the core test set (192 files for the TIMIT test set), for 8 mixtures in the HMM output probability density, a percentage correct of 65.18%, an accuracy of 62.38%, 4703 hits, 1006 deletions, 1506 substitutions, and 202 insertions is obtained, for a total of 7215 phonemes in the 192 sentences. This can be compared to the case when no language model or word transition penalty is used, which gives a percentage correct of 63.92%, an accuracy of 52.99%, 4612 hits, 651 deletions, 1952 substitutions, and 789 insertions on the core test set. The language model scale factor is found to give the best results at 4.0 when no word transition penalty is used.

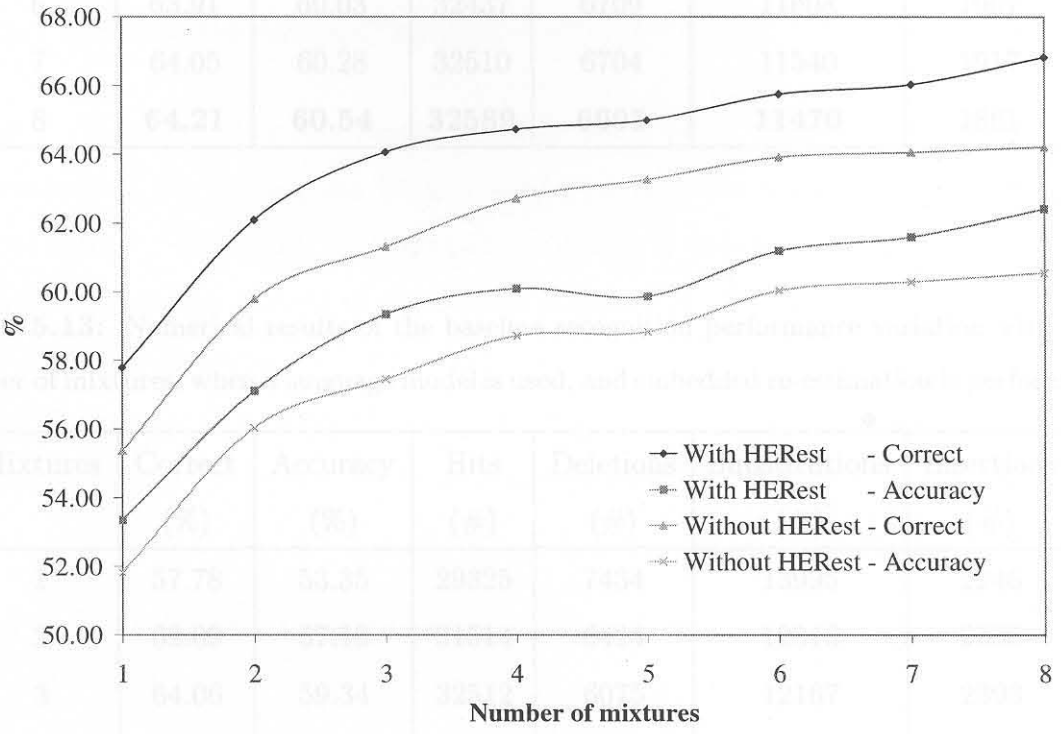

Figure 5.7: Effect of the number of mixtures on HMM phoneme recognition performance. A language model is used.

Table 5.12: Numerical results of the baseline recognition performance variation with the number of mixtures, when a language model is used, and no embedded re-estimation is performed.

Mixtures	Correct (%)	Accuracy (%)	Hits (#)	Deletions (#)	Substitutions (#)	Insertions (#)
1	55.37	51.83	28104	8655	13995	**1800**
2	59.78	56.04	30341	7615	12798	1897
3	61.31	57.44	31115	7185	12454	1962
4	62.71	58.71	31827	6861	12066	2031
5	63.26	58.83	32105	6790	11859	2246
6	63.91	60.03	32437	6709	11608	1967
7	64.05	60.28	32510	6704	11540	1917
8	**64.21**	**60.54**	**32589**	**6695**	**11470**	1861

Table 5.13: Numerical results of the baseline recognition performance variation with the number of mixtures, when a language model is used, and embedded re-estimation is performed.

Mixtures	Correct (%)	Accuracy (%)	Hits (#)	Deletions (#)	Substitutions (#)	Insertions (#)
1	57.78	53.35	29325	7434	13995	2246
2	62.09	57.10	31514	6424	12816	2535
3	64.06	59.34	32512	6075	12167	2393
4	64.73	60.09	32855	5949	11950	2355
5	64.99	59.86	32984	5905	11865	2605
6	65.75	61.19	33372	5783	11599	2318
7	66.03	61.59	33514	5821	11419	2254
8	**66.82**	**62.41**	**33914**	**5685**	**11155**	**2237**

5.2.3 Word transition penalty

Purpose: The purpose of this experiment is to determine the effect that a word transition penalty has on the phoneme recognition performance of an HMM based recognition system when a language model is not used.

Experimental setup: Speech utterances are recognised using trained HMMs. The HMMs are trained on 3696 files of the TIMIT training set, and evaluated by recognising the 192 files of the TIMIT test set (the core test set). A language model is not used. In order to determine the "optimal" value of the fixed word transition penalty (the bias term, w_b, in Equation (4.9)), experiments were performed with w_b varied from -10.0 to 20.0 in steps of 1.0. The word transition penalty that results in the highest recognition accuracy is seen as the "optimal" value.

Results: When no language model is used, but a word transition penalty is, the performance of the system can be increased over the case when no word transition penalty is used. When the word transition penalty is optimised on the TIMIT core test set, a percentage correct of 60.18%, an accuracy of 55.45%, 4342 hits, 1052 deletions, 1821 substitutions, and 341 insertions on the core test set (containing a total of 7215 phonemes). The word transition penalty is found to give the best recognition performance when set to -6.0, when no language model is used.

5.2.4 Combined language model and word transition penalty

Purpose: The purpose of this experiment is to determine the effect that the combined use of a language model and word transition penalty has on the phoneme recognition performance of an HMM based recognition system.

Experimental setup: Speech utterances are recognised using trained HMMs. The HMMs are trained on 3696 files of the TIMIT training set, and evaluated by recognising

the 192 files of the TIMIT core test set. A bigram language model is used. In order to determine the "optimal" values of the language model scale factor (s in Equation (2.45)) and fixed word transition penalty (the bias term, w_b, in Equation (4.9)), they are varied from 0.0 to 10.0, and -10.0 to 20.0, respectively, in steps of 1.0. The values that result in the highest recognition accuracy is seen as the "optimal" values.

Results: When a language model is used, as well as a word transition penalty, the performance of the system can be increased significantly. When the language model scale factor and word transition penalty is jointly optimised on the TIMIT core test set, a percentage correct of 67.05%, an accuracy of 63.23%, 4838 hits, 820 deletions, 1557 substitutions, and 276 insertions on the core test set (containing a total of 7215 phonemes). The language model scale factor and word transition penalty is found to give the best recognition performance when both are set to 5.0. Figure 5.8 shows the general trend of the recognition performance, as the language model scale factor is varied, with the fixed word transition penalty (bias) fixed at 17.0. Figure 5.9 shows the general trend of the recognition performance, as the fixed (bias) word transition penalty is varied, with the language model scale factor fixed at 4.0.

5.3 Experiment 3: Speech recognition using segmentation information

This section reports results on the recognition of speech using segmentation information, as discussed in detail in Chapter 4. Results on the modification of the HMM transition probabilities (both linear and non-linear), as well as an adaptive word transition penalty, are given. The combined effect of transition and word transition penalty modification is also evaluated.

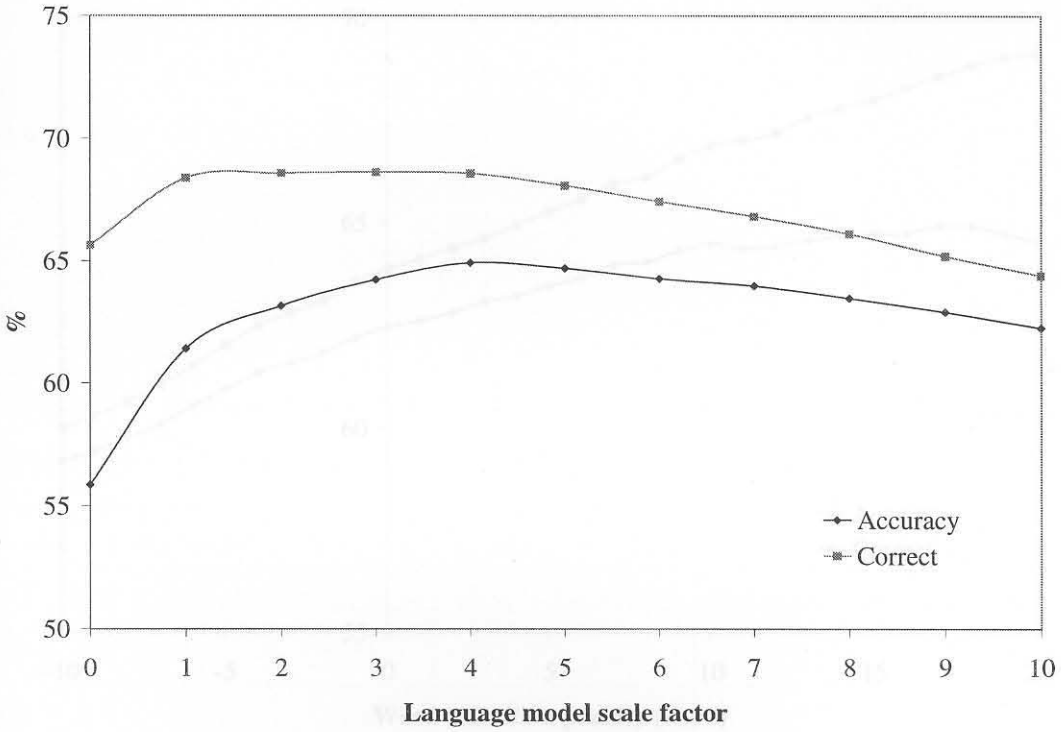

Figure 5.8: Effect of the language model scale factor on HMM recognition performance. A fixed (bias) word transition penalty of 17.0 is used.

5.3.1 HMM transition probability modification

Purpose: The purpose of this experiment is to determine the effect that the modification of HMM transition probabilities, from the segmentation information, has on the phoneme recognition performance of an HMM based recognition system.

Experimental setup: Speech utterances are recognised using trained HMMs. The HMMs are trained on 3696 files of the TIMIT training set, and evaluated by recognising the 192 files of the TIMIT test set (the core test set). When a language model is used, a bigram language model is chosen, and the word transition penalty used is just the fixed bias term, as described in Chapter 4. In order to determine the "optimal" values of the language model scale factor (s in Equation (2.45)) and fixed word transition penalty (the bias term, w_b, in Equation (4.9)), they are varied from 0.0 to 10.0, and -10.0 to

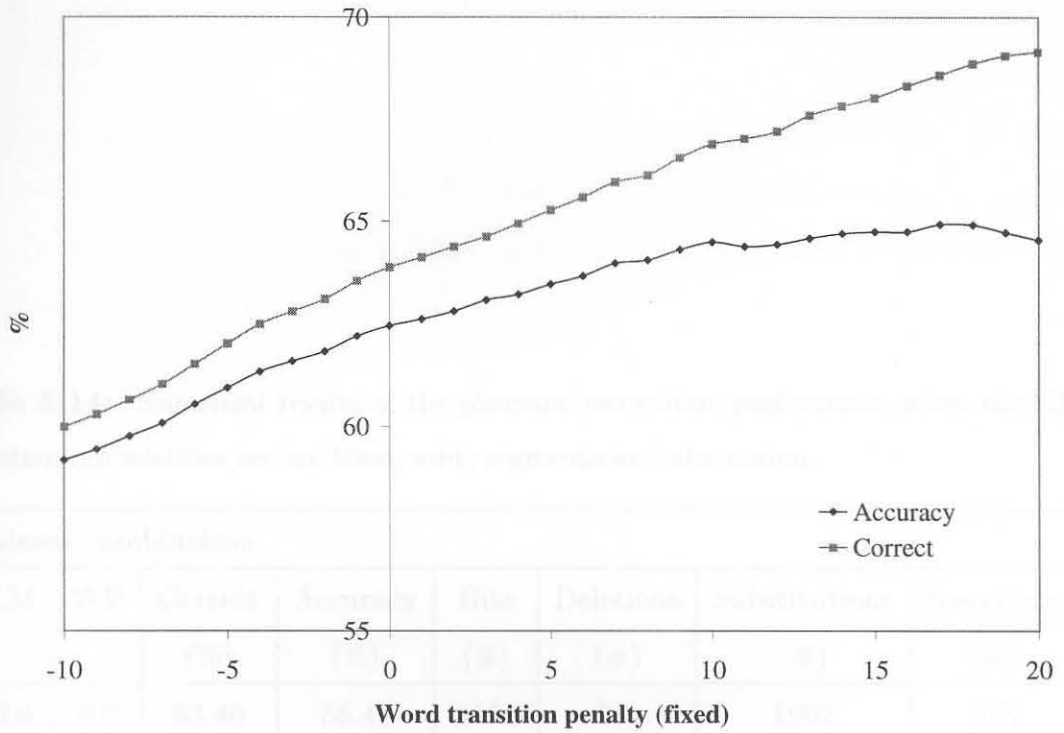

Figure 5.9: Effect of the fixed (bias) word transition penalty on HMM recognition performance. A language model scale factor of 4.0 is used.

20.0, respectively, in steps of 1.0. The values that result in the highest recognition accuracy is seen as the "optimal" values.

Results: The use of segmentation information, to modify the HMM transition probabilities, can increase the recognition performance of the system. Table 5.14 gives the results numerically. In this table, "LM" indicates the language model scale factor and "WP" indicates the fixed word transition penalty (bias). Results are given for both linear and non-linear combination of segmentation information with HMM transition probabilities.

Table 5.14: Numerical results of the phoneme recognition performance when the HMM transition probabilities are modified, using segmentation information.

Linear combination							
LM	WP	Correct (%)	Accuracy (%)	Hits (#)	Deletions (#)	Substitutions (#)	Insertions (#)
0.0	0.0	63.40	55.41	4574	739	1902	576
0.0	-4.0	61.00	56.34	4401	989	1825	336
3.0	0.0	65.29	62.54	4711	963	**1541**	**199**
5.0	8.0	**67.78**	**63.58**	**4890**	**730**	1595	303
Non-linear combination							
LM	WP	Correct (%)	Accuracy (%)	Hits (#)	Deletions (#)	Substitutions (#)	Insertions (#)
0.0	0.0	64.27	52.47	4637	**620**	1958	851
0.0	-5.0	61.29	55.56	4422	934	1859	413
4.0	0.0	65.36	62.37	4716	976	**1523**	**216**
5.0	5.0	**67.08**	**63.02**	**4840**	805	1570	293

5.3.2 HMM word transition penalty modification

Purpose: The purpose of this experiment is to determine the effect that the modification of HMM word transition penalty (used here as a phoneme transition penalty), from the segmentation information, has on the phoneme recognition performance of an HMM based recognition system.

Experimental setup: Speech utterances are recognised using trained HMMs. The HMMs are trained on 3696 files of the TIMIT training set, and evaluated by recognising the 192 files of the TIMIT test set (the core test set). When a language model is used, a bigram language model is chosen, and the word transition penalty used is the fixed bias term, plus two additional time-varying components, as described in Chapter 4. In order to determine the "optimal" values of the language model scale factor (s in Equation (2.45)) and fixed word transition penalty (the bias term, w_b, in Equation (4.9)), they are varied from 0.0 to 10.0, and -10.0 to 20.0, respectively, in steps of 1.0. The values that result in the highest recognition accuracy is seen as the "optimal" values. The scale factors (a and b) for the two time-varying components (w_p and w_c in Equation (4.9)) of the word transition penalty are set equal to each other and varied from 0.0 to 15.0 in steps of 1.0, in order to determine the "optimum" scale factor.

Results: The use of segmentation information, to modify the HMM word transition penalty term, can increase the recognition performance of the system. Table 5.15 gives the results numerically. In this table, "LM" indicates the language model scale factor, "WP" indicates the fixed word transition penalty (bias), and "AWP" indicates the scale factors ((a and b in Equation (4.9)) used for the two time-varying word transition penalty terms (w_c and w_p in Equation (4.9)). Figure 5.10 shows the effect that the adaptive word transition penalty terms have on the recognition performance, when both a language model and fixed word transition penalty term are used.

Table 5.15: Numerical results of the phoneme recognition performance when the word transition penalty is modified, using segmentation information.

LM	WP	AWP	Correct (%)	Accuracy (%)	Hits (#)	Deletions (#)	Substitutions (#)	Insertions (#)
0.0	0.0	9.0	61.98	59.40	4472	859	1884	186
0.0	1.0	9.0	62.33	59.60	4497	819	1899	197
3.0	0.0	3.0	65.07	63.01	4695	963	**1557**	**149**
4.0	17.0	10.0	**68.58**	**64.93**	**4948**	**610**	1657	263

5.3.3 Combined transition probability and word penalty modification

Purpose: The purpose of this experiment is to determine the effect that the combined modification of HMM transition probabilities and word transition penalty (used here as a phoneme transition penalty), from the segmentation information, has on the phoneme recognition performance of an HMM based recognition system.

Experimental setup: Speech utterances are recognised using trained HMMs. The HMMs are trained on 3696 files of the TIMIT training set, and evaluated by recognising the 192 files of the TIMIT core test set. When a language model is used, a bigram language model is chosen, and the word transition penalty used is just the fixed bias term, as described in Chapter 4. In order to determine the "optimal" values of the language model scale factor (s in Equation (2.45)) and fixed word transition penalty (the bias term, w_b, in Equation (4.9)), they are varied from 0.0 to 10.0, and -10.0 to 20.0, respectively, in steps of 1.0. The values that result in the highest recognition accuracy is seen as the "optimal" values. The scale factors (a and b) for the two time-varying components (w_p and w_c in Equation (4.9)) of the word transition penalty are set equal to each other and varied from 0.0 to 15.0 in steps of 1.0, in order to determine

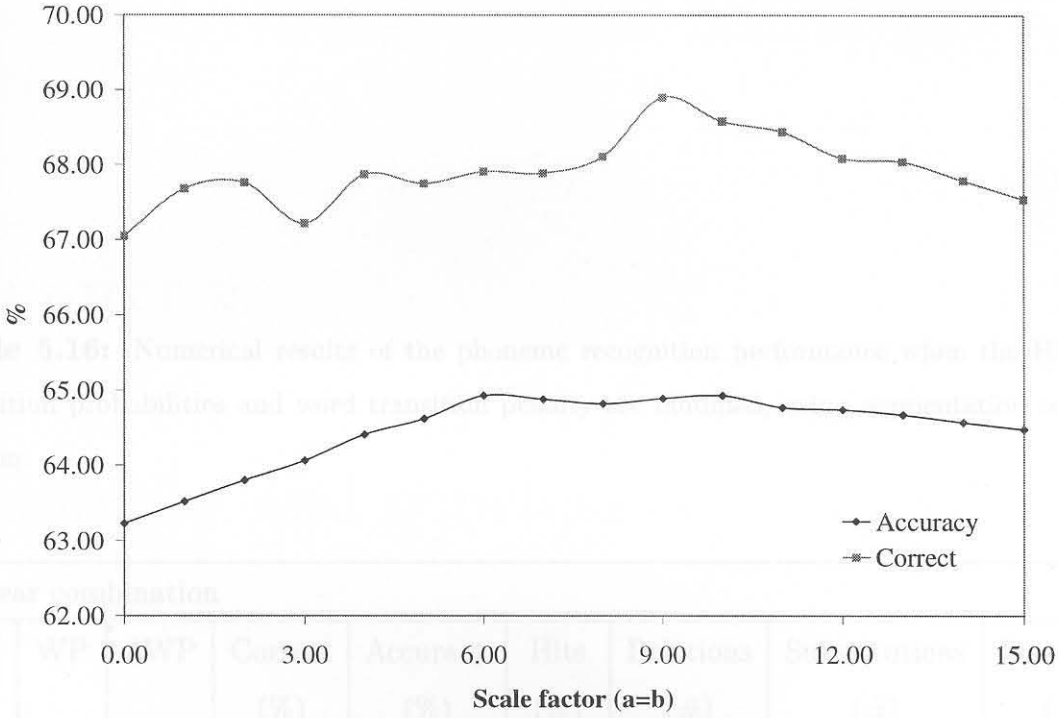

Figure 5.10: Effect of the adaptive word transition penalty scale factor on the phoneme recognition performance.

the "optimum" scale factor.

Results: The use of segmentation information, to modify the HMM transition probabilities, as well as the word transition penalty term, can increase the recognition performance of the system. Table 5.16 gives the results numerically. In this table, "LM" indicates the language model scale factor, "WP" indicates the fixed word transition penalty (bias), and "AWP" indicates the scale factors (a and b in Equation (4.9)) used for the two time-varying word transition penalty terms (w_c and w_p in Equation (4.9)). Results are given for both linear and non-linear combination of segmentation information with HMM transition probabilities.

Table 5.16: Numerical results of the phoneme recognition performance when the HMM transition probabilities and word transition penalty are modified, using segmentation information.

Linear combination								
LM	WP	AWP	Correct (%)	Accuracy (%)	Hits (#)	Deletions (#)	Substitutions (#)	Insertions (#)
0.0	0.0	8.0	61.66	59.22	4449	882	1884	**176**
0.0	3.0	9.0	62.56	59.58	4514	785	1916	215
3.0	0.0	1.0	65.03	62.73	4692	986	**1537**	166
4.0	17.0	8.0	**68.84**	**64.91**	**4967**	**597**	1651	284
Non-linear combination								
LM	WP	AWP	Correct (%)	Accuracy (%)	Hits (#)	Deletions (#)	Substitutions (#)	Insertions (#)
0.0	0.0	9.0	61.91	59.21	4467	841	1907	195
0.0	5.0	11.0	62.80	59.33	4531	727	1957	250
3.0	0.0	3.0	65.35	63.10	4715	943	**1557**	**162**
4.0	15.0	9.0	**68.30**	**64.80**	**4928**	**616**	1671	253