Chapter 4

Duration Modeling - Part I

Duration models are developed for predicting the duration of speech segments used for speech synthesis. The duration models take feature vector as input and assigns duration value to each. From the duration analysis on Malayalam speech, it has been seen that the following factors are statistically significant.

- \( F_1 \) : Phoneme identity ; 69 levels
- \( F_2 \) : Position of the phoneme in the word ; 4 levels
- \( F_3 \) : Identity of preceding phoneme ; 69 levels
- \( F_4 \) : Identity of succeeding phoneme ; 69 levels
- \( F_5 \) : Position of the word in the sentence ; 5 levels
- \( F_6 \) : Number of phonemes in the word ; 10 levels

These factors constitute the feature vector \( \vec{f} \), which can be used for prediction.
This chapter reports the development of two duration models - i) probabilistic duration model and ii) hybrid duration model using CART and HMM. The duration models were experimented on a speech database of Malayalam news. The training database included a total of 29404 phonemes, 3075 words and 344 sentences (approximately 75% of the database). The models were tested on a database consisting of 10064 phonemes, 1155 words and 118 sentences (approximately 25% of the database). The feature vectors for all the phonemes in the database are found, which is given to the duration models, along with duration values.

### 4.1 Probabilistic duration model

The probability distribution of duration values after fixing different factors (Figure 3.36) has shown that, as we go on fixing the levels of the factors, the variance decreases. The standard deviation has shown a reduction from 17.2 ms to 6.1 ms (Figure 3.37). Hence the mean value of duration can be used as an approximate estimate of duration, for the phonemes represented by that combination of factor values (feature vector). To get the duration value corresponding to each feature vector, the duration values are to be grouped according to the levels taken by the different factors \((f_1, \ldots, f_n)\). The probability density function (pdf) corresponding to each feature vector \(f = f_1, \ldots, f_n\) is to be obtained, for duration prediction. The probability density function of duration corresponding to each feature vector \(f = f_1, \ldots, f_n\), is the conditional
distribution of duration \( d \) given \( f \) \((E(p(d/f = f_1,..f_n)))\) \(^1\). The expectation (mean) of the distribution can be used to represent the duration of phonemes for that particular feature vector.

\[
\text{Predicted duration} = E(p(d/f = f_1,..f_n))
\]

When the duration of phonemes are predicted for each feature vector in this manner, the standard deviation corresponds to the root mean square error (RMSE), which is the objective measure used for evaluating performance of duration model. If sufficient number of factors are analyzed, it is possible to predict duration of phonemes.

But as the number of factors increase, the number of such groups increases exponentially and it is impractical to obtain expectation values for all such groups\(^2\). To have sufficient number of data in each group, the database has to be prohibitively large.

This process of finding conditional probability can be simplified if the occurrence of factors \( f_1, f_2, ... f_n \) are independent. If the components of the random vector \( f \) are assumed to be independent random variables, then the conditional probability \( p(d/f_1, f_2, f_3...f_n) \) can be obtained from the conditional probabilities of duration with respect to each factor \((p(d/f_1),...p(d/f_n))\)

\[
p(f_1/d) = \frac{p(d/f_1) \cdot p(f_1)}{p(d)}
\]

\(^1\)Duration is a random variable and feature vector is a random vector.
\(^2\)If the feature vector consists of 4 factors, say phoneme identity (69 levels), position of the phoneme in the word (10 levels), identity of preceding phoneme (69 levels) and identity of succeeding phoneme (69 levels) the number of distinct feature vectors and groups comes to 3285090 \((69 \times 10 \times 69 \times 69)\)
\[ p(f_n/d) = \frac{p(d/f_n) p(f_n)}{p(d)} \]  \hspace{1cm} (4.4)

If the random variables \( f_1 \) and \( f_2 \) are independent,

\[ p(f_1 f_2/d) = p(f_1/d) p(f_2/d) \]  \hspace{1cm} (4.5)

\[ p(f_1 f_2/d) = \frac{p(d/f_1) p(f_1) p(d/f_2) p(f_2)}{p(d)^2} \]  \hspace{1cm} (4.6)

Similarly,

\[ p(f_1 f_2 \ldots f_n/d) = \frac{\prod_{i=1}^{n} p(d/f_i) p(f_i)}{p(d)^n} \]  \hspace{1cm} (4.7)

Applying Bayes theorem,

\[ p(d/f_1, f_2, \ldots f_n) = \frac{p(f_1, f_2, \ldots f_n/d) p(d)}{p(f_1, f_2, \ldots f_n)} \]  \hspace{1cm} (4.8)

Substituting the value of \( p(f_1 f_2 \ldots f_n/d) \) from equation 4.7:

\[ p(d/f_1, f_2, \ldots f_n) = \frac{\prod_{i=1}^{n} p(d/f_i) p(f_i) p(d)}{p(d)^n p(f_1, f_2 \ldots f_n)} \]  \hspace{1cm} (4.9)

Since independence is assumed,

\[ p(f_1, f_2 \ldots f_n) = p(f_1) p(f_2) \ldots \ldots p(f_n) \]  \hspace{1cm} (4.10)

\[ p(d/f_1, f_2, \ldots f_n) = \frac{\prod_{i=1}^{n} p(d/f_i)}{[p(d)]^{n-1}} \]  \hspace{1cm} (4.11)

The conditional probability of duration \( d \) given a feature vector \( f \) can
thus be determined from the conditional probabilities of duration given each factor, that is $p(d/f_i)$. The number of conditional distributions $p(d/f_i)$ to be evaluated, now reduces to sum of the number of different values taken by the factors (for example, if $f_i$ takes 8 values, 8 distributions are to be evaluated for that factor). For each feature vector $f$, the conditional distribution of duration $d$ given that feature vector $p(d/f)$ can be evaluated using equation 4.11. From the conditional distribution, the duration value that gives the maximum value of probability can be found. The value of duration that maximizes the probability $p(d/(f_1, f_2, f_3, ..., f_n))$ that is $p(d/f)_{max}$ can be used to predict the value of duration for that feature vector. A small amount of gaussian noise ($\epsilon$) can be added to this duration value to reflect the high degree of variability in speech production [104]. Hence the predicted duration can be written as

$$\text{predicted duration} = \text{duration corresponding to } p(d/f)_{max} + \epsilon \quad (4.12)$$

The duration model developed using this concept is schematically depicted in Figure 4.1. The model has a multi-layer structure, with each layer having states corresponding to the number of levels taken by each factor. The different states in a layer corresponds to the different levels taken by the factor. At each state, the emitted probabilities are the conditional probabilities at that layer for different duration intervals. The duration being a continuous random variable, probability is not defined at each value. The probability for a range of values can be evaluated from probability density function.
Hence the total range of duration is divided into small intervals. At layer 1 only one factor is considered. The predicted duration corresponds to that factor alone. As we proceed to the higher levels, more factors will be taken into account. Each layer adds one more factor to the feature vector. The advantage of this structure is that, more factors can be added to the model easily without disturbing\footnote{The conditional probabilities $p(d/f)$ at a layer is not altered, when more levels are added subsequently.} the existing structure.

The parameters of the model consists of the conditional probabilities of duration with respect to different values of feature vector. The evaluation of these values constitute the training of the model. In the training phase, the conditional pdf of the duration of the vowels ($p(d/f_i)$) with respect to each factor $f_i$ is found. The total range of duration values is divided into small intervals of 1ms and the probability values for each interval is evaluated from the pdf and is stored in a look up table. If $f_i$ takes on M levels, the conditional pdf $p(d/f_i)$ is found for each of the M values taken by $f_i$.

When a feature vector is given for prediction of duration, the model traverses a particular path, corresponding to the values taken by each factor of the feature vector. At each layer, the model occupies the state, corresponding to the level taken by that factor. The conditional pdf till that point is evaluated using equation 4.11. The conditional pdf for the given feature vector is obtained, as the model reaches the final layer. From the probability values, the duration range that corresponds to the maximum value of probability is found out. This value is modified by adding gaussian noise $\epsilon$ (equation 4.12).

As an example, consider a feature vector having 3 factors, $f_1, f_2, f_3$ given to the duration model. Suppose the value of feature vector given for predic-
Figure 4.1: Schematic diagram of the probabilistic model
tion is \((2, 1, 10)\). The model evaluates the conditional pdf

\[ p(d/f) = p(d/f_1 = 2) \times p(d/f_2 = 1) \times p(d/f_3 = 10) /[p(d)]^2 \]

If the duration \(d\) for this phoneme ranges from 10 ms to 100 ms, with intervals of 1 ms, the conditional probabilities \(p(d/f)\) for each of these 91 intervals are evaluated. The duration interval that gives the maximum probability value, modified by the gaussian noise, forms the predicted duration.

The root mean square obtained when the model was 13.2 ms and the correlation was 0.80. The mean opinion score was 2.91. The limitation of this model is that there is not enough evidence to prove that the factors \(f_1, f_2, \ldots, f_n\) are independent random variables. A set of random variables will be independent, only if it is so for all combinations.

### 4.2 Classification and regression trees

Non linear regression methods like Classification and Regression Tree can be used to predict the duration according to different factors. The CART can classify data according to different factors and regress the values corresponding to each feature vector. The CART model for Malayalam speech is developed using tree fitting function in MATLAB\(^5\).

The tree is formed by giving the duration values and the corresponding feature vectors. The basic building algorithm starts with a set of feature vectors and at each stage, asks all possible questions for all possible features, to find out how to split data, so that the variance in each subgroup is the

\(^5\)Another commonly used tool for CART tree building is WAGON
least amongst all such binary partitions.

The basic CART building algorithm is a greedy algorithm\(^6\). It chooses the locally best discriminatory feature at each stage in the process. This is suboptimal, but a full search for a fully optimized set of question would be computationally very expensive. If the data set has outliers, some of the lower branches may be affected by this. Hence a tree like this, having many branches, may fit the current data set well but would not be so at predicting new values. The problem of overfitting\(^7\) can be minimised if we could arrive at a simpler tree, where some of the leaf nodes will be removed and corresponding branch nodes will be converted into leaf nodes. This process called pruning, provides a more generalized or abstract version reducing the problem of overfitting. At each level of pruning, one branch node will be turned to leaf node. The figure 4.2 gives a portion of the CART of phoneme /u/, pruned to level 205.

For testing the tree, the cost of the tree can be measured using resubstitution method. The cost of a node is the average squared error over the observations in that node. The cost of the tree is the sum over all terminal nodes of the estimated probability of that node, times the node’s cost. In MATLAB, the tree testing function using resubstitution method, returns a vector of cost values for each subtree in the optimal pruning sequence for the tree. The cost of CART, for phoneme /k/ is in Figure 4.3.

The best tree size can be estimated using cross validation procedure.

---

\(^6\)A greedy algorithm always takes the best possible decision (not all feasible decisions) at each stage, based on the current local optimum and the best decision (not all possible decisions) made in the previous stage. It is not exhaustive and may not give accurate answers in certain problems.

\(^7\)Since branch nodes split into children nodes for all data in the training set including outliers, it may carry out some insignificant classification as well (in other words it does overfitting for current data).
Figure 4.3: Cost of CART for phoneme /k/

Figure 4.4: Cost of the CART for phoneme /a/
Cross-validation is the statistical practice of partitioning a sample of data into subsets such that the analysis is initially performed on a single subset, while the other subsets are retained for subsequent use in confirming and validating the initial analysis. In K-fold cross validation, the original sample is partitioned into K subsamples. Of the K subsamples, a single subsample is retained as the validation data for testing the model, and the remaining K-1 subsamples are used as training data. The cross validation process is then repeated K times, with each of the K subsamples used exactly once as the validation data. The K results from the folds can then be averaged to produce a single estimation. 10 fold cross validation is commonly used. In a 10-fold cross-validation procedure, the data set will be partitioned into 10 subsamples, chosen randomly but with roughly equal size. For each subsample, the tree is fitted to the remaining data and uses it to predict the subsample. It pools the information from all subsamples to compute the cost for the whole
The best tree is the sub tree which gives minimum value of cost. The MATLAB function for tree testing using cross validation procedure is used to find the best level of pruning.

Figure 4.4 gives the cost of the subtrees using resubstitution and cross validation procedures for phoneme /a/. The pruning level for obtaining best tree is returned by the function was 223. The tree of phoneme /a/ pruned to level 223 is given in Figure 4.5.

The disadvantages of CART are:

1. The CART predicts duration according to the feature vector comprising of factors that can be directly determined from text. The duration patterns of speech depends upon a large number of factors. All these factors cannot be determined from text. The duration analysis had shown that, even when the different factors are fixed, the duration values show variations, which indicate that, the duration cannot be predicted beyond a certain extent, using factors that are determined from text.

2. The CART fails in predicting values for rare feature vectors. It does not have an interpolating property. If a non linear regression model like CART is combined with a stochastic model, the variation due to other interacting factors can also be captured to a certain extent. The hybrid duration model combining CART and HMM is an attempt in this direction.
4.3 Hybrid duration model using CART and HMM

Speech being a continuous process, the prosodic parameters of neighboring phonemes interact with each other. The duration values depend on a large number of factors which cannot be easily extracted from text, like complex interaction between phonetic segments, syntactic and semantic factors etc. Only a few factors can be identified by statistical analysis of duration values. The variations due to many of these can be incorporated by combining CART model with a stochastic model like Hidden Markov Model (HMM).

4.3.1 Hidden Markov Model

Hidden Markov Models are stochastic models where the system being modeled is assumed to be a Markov process\(^8\) with hidden states. It consists of a finite set of states, each of which is associated with a probability distribution. Transitions among the states are governed by a set of probabilities called transition probabilities. In a particular state an outcome or observation can be generated, according to the associated probability distribution known as emission probabilities. In HMM, only the outcome is visible and the states are hidden. The HMM is thus a doubly embedded stochastic process with an underlying stochastic process that is not observable (hidden), but can only be observed through another set of stochastic process, that produce the sequence of observations [143]. An HMM is characterized by the

---

\(^8\)A Markov process is a stochastic process, that uniquely determine the future behaviour of the process by its present state. The distributions of events are independent of the history of the system.
The following parameters [143].

1. The number of states of the model, \( N \). The set of states are denoted as \( S = \{ S_1, S_2, \ldots, S_N \} \) and the state at time \( t \) is \( q_t \).

2. The number of distinct observation symbols per state, \( M \) (discrete alphabet size). The individual symbols are denoted as \( V = \{ v_1, v_2, \ldots, v_M \} \).

3. The state transition probability distribution \( A = \{ a_{ij} \} \).

\[
a_{ij} = P[q_{t+1} = S_j | q_t = S_i], \quad 1 \leq i, j \leq N
\]

4. The observation symbol probability distribution in state \( j \), \( B = \{ b_j(k) \} \), where

\[
b_j(k) = P[v_k \text{ at } t | q_t = S_j], \quad 1 \leq j \leq N, \quad 1 \leq k \leq M
\]

5. The initial state distribution \( \pi = \{ \pi_j \} \), where

\[
\pi_i = P[q_1 = S_i], \quad 1 \leq i \leq N
\]

The HMM \( \lambda = (A, B, \pi) \) generates an observation sequence

\[
O = O_1 O_2 \ldots O_T
\]

in accordance with the values of \( A, B \) and \( \pi \). \( O_t \) is one of the symbols from \( V \) and \( T \) denotes the number of observations in the sequence.

The three basic problems handled by HMM are

- **Evaluation**: Given \( O = O_1 O_2 \ldots O_T \) and \( \lambda = (A, B, \pi) \), compute \( P(O|\lambda) \) the probability of the observation sequence, given the model.
• Decoding: Given \( O = O_1 O_2 ... O_T \) and \( \lambda = (A, B, \pi) \), find the state sequence \( Q = q_1 q_2 ... q_T \) that best explains the observation.

• Learning: How to adjust the model parameters \( \lambda = (A, B, \pi) \) to maximize \( P(O|\lambda) \) ?

The different problems are addressed according to application. Evaluation can be used to choose the model that best matches the observation among several competing models. Decoding uncovers the hidden part of the model, by finding the optimal state sequence for the given observation as required by applications like continuous speech recognition. In speech recognition, the sequence of speech constructs (states) is to be found, from given observations (features extracted from speech wave form). Learning is required in all applications to optimally adapt model parameters to observed training data.

The type\(^9\) and architecture of HMM depends on the application for which it is designed for. For speech related applications, left-right models find useful. HMM has been extensively used in speech recognition applications \([58, 60, 67, 66]\). HMM based speech synthesis is reported in literature \([57, 59, 62]\). In most of these models, states represent speech segments and the duration is modeled using gaussian distribution or using context tree based clustering.

The duration model using HMM for prediction of duration corresponding to a given text is new in literature. The HMM proposed in this work has a multilevel architecture. The phonemes in a word are considered as the observations and the hidden parameter correspond to duration. For the given

\(^9\)Types of HMM: Ergodic (every state can be reached from every other state in a finite number of states, left-right model (states proceed from left to right) or variants of these
sequence of phonemes, the state sequence is found which is used to predict the
duration of phonemes. This model deals with decoding problem for duration
prediction and learning problem for finding the model parameters.

4.3.2 Duration model

The hybrid duration model is a combination of CART and HMM. The CART
model predicts duration corresponding to the feature vectors describing the
phonemes in the training data base. The difference between the actual dura­
tion value and the value predicted by CART is calculated. These deviation
values form the training data base for HMM. From the histogram of the
deviation values (figure 4.6), it can be seen that the deviation values range
mostly between -20ms to +20ms (90%) with extreme values coming up to
±50ms. The deviation ranging from -21 to +21 is divided into 15 intervals
with a step value of 3. Each of these intervals form a particular state of each
layer of HMM.

The HMM has 16 levels, with each level corresponding to a phoneme in the
word. The number of levels was chosen based on the number of phonemes in
words in the database. From the Figure 4.7 showing the histogram of number
of phonemes in words, it can be seen that very few words have number of
phonemes greater than 16 (only 99.04% of words have number of phonemes
less than 16). Hence the number of levels is taken as 16.

In each level 15 states are there, with each state corresponding to a par­
ticular interval of deviation (say 1ms to 3ms). The structure of HMM for
duration modeling is shown in Figure 4.8. In this HMM the deviation val­
ues are the hidden states and phonemes are the observations emitted by the
Each state emits probabilities of different phonemes occupying that particular state. The initial probability of the different states denotes the probability of the first phoneme occupying that particular state. If /a/ is the first phoneme, the initial probabilities of each state is multiplied with the emission probability of /a/ in that state. The state that gives the maximum value for this will be the initial state. Each state in a level can move to any state in the next level or to the end state, depending upon the number of phonemes in the word. The transition probability denotes the probability of transition from a state in a level to the different states in the next layer or to the end state. The transition probability from state i in level k to state j in level k+1, can be interpreted as the probability of the \((k + 1)^{th}\) level phoneme occupying state j (deviation in the \(j^{th}\) interval) given that the \(k^{th}\)
Figure 4.7: Histogram of number of phonemes in words

level phoneme is in state i (deviation in the \(i^{th}\) interval). These can capture interaction between phonemes as well as the factors which affect the word as a whole.

The training of the model involves finding \(A, B\) and \(\pi\). The probabilities are predicted using the database as given below.

\[
\pi_i = \frac{(Number \ of \ times \ in \ state \ S_i \ at \ time \ t = 1) + 1}{(Total \ number \ of \ values \ in \ the \ data \ set) + 15}
\]

\[
a_{ijm} = \frac{(Number \ of \ times \ in \ state \ S_i \ in \ level \ m \ to \ state \ S_j \ in \ level \ m + 1) + 1}{(Number \ of \ transitions \ from \ level \ m \ to \ level \ m + 1) + 15}
\]

\[
b_{jm}(k) = \frac{(Number \ of \ times \ phoneme \ k \ is \ emitted \ in \ state \ j \ in \ level \ m) + 1}{(Number \ of \ emissions \ at \ level \ m) + 69}
\]
Figure 4.8: Hidden Markov Model for duration modeling

97
Figure 4.9: A portion of duration model

Level 5

i5  
(9ms - 11ms)

Level 6

i6  
(9ms - 11ms)

j6  
(12ms - 15ms)
The constants in the numerator and denominator are to ensure that the probabilities never go to zero. Otherwise while finding the best path, if any one of the probabilities happen to be zero, the whole path will be eliminated. With a small constant in the numerator and a large constant in the denominator, a very small value of probability results instead of zero. The HMM is specified by $\lambda = (A, B, \pi)$, where the initial probability vector $\pi$ is of dimension $1 \times 15$, transition probability vector $A$ is of dimension $16 \times 15 \times 15^{10}$ and emission probability vector $B$ is of dimension $16 \times 15 \times 69^{11}$.

When the phonemes corresponding to a word are given as input, at each level, the emission probability of each phoneme in each state, is multiplied with transition probabilities to the next level. The best path is found using viterbi algorithm, yielding the sequence of states for the given word. Since each state correspond to deviation of duration values from that predicted by CART the best path gives the deviation of duration values for the phonemes in the word. The mid value of the deviation in the intervals corresponding to the state sequence obtained, is added with that predicted by CART to form the final predicted value from the hybrid model $^{12}$. For the combined model, the root mean square error (RMSE) was 8.32 ms, correlation was 0.93 and mean opinion score was 3.12.

$^{10}$16 levels; 15 states in each level; 15 transitions in each state  
$^{11}$16 levels; 15 states in each level; 69 emissions in each state  
$^{12}$If the interval corresponding to a state in the best path obtained is 10ms to 12ms, the mid value of the interval 11ms will be taken for addition with duration predicted by CART
4.4 Summary of results

Two new models were developed using the conventional approach of considering each speech unit (phoneme) separately, analyzing the factors affecting duration, and forming the feature vectors based on this. The feature vectors along with the corresponding duration values form the training data for the models.

- The probabilistic duration model was developed based on the conditional probability distribution of the phonemes, given different factor values. The predicted duration is that which gives maximum value for the conditional probability, given the feature vector. The model gave a root mean square value of 13.2 ms, correlation of 0.80 and mean opinion score of 2.91.

- In the hybrid model CART classifies the data according to the factors that can be determined from text and HMM captures the variation due to complex interaction of factors and due to factors that are not easily extractable from text. The hybrid model incorporates the advantages of both CART and HMM. Since finding the best path involves all the phonemes in the word, the rhythm of the word as a whole can be captured to an extent. The root mean square error (RMSE) of the combined model was 8.32 ms, correlation was 0.93 and mean opinion score was 3.12.