A New Fangled Insinuation for Stress Affect Speech Classification

Nachamai.M
Dept. of MCA
Alliance Business Academy

T. Santhanam
PG & Research Dept,
DG Vaishnav college

C.P.Sumathi
Dept. of Computer science,
SDNB Vaishnav college

ABSTRACT
Demarcation in human interaction is through two channels: one transmits explicit messages; the other transmits implicit messages about the speakers themselves knowingly or unknowingly. Both linguistics and technology have invested enormous effort in trying to understand the first (explicit) channel, but the second (implicit) is not as well understood. First, building an emotion detection system makes it possible to assess the extent to which theoretical proposals explain people's everyday competence in understanding emotion. Second, model building enforces coherence. It is true that emotions play an important role in the making of speech. The deduction of emotions from speech is of recent origin and it is the primary focus of this research paper.

Keywords
Affect Recognition, Speech analysis, Support vector machine (SVM), Probabilistic Neural Network(PNN), Hidden Markov Model(HMM).

1. INTRODUCTION
Human beings express thoughts, feelings and ideas orally to one another through a series of complex movements that alter and mold the basic tone created by voice into speech/decodable sounds. In colloquial terms speech is a linguistic act designed to convey information.

Affect in speech
Etymologically, the word *emotion* is a composite of two Latin words, Ex/out, outward + motio/movement, action or gesture. Affective computing is computing that relates to, arises from or deliberately influences emotion. It helps communicate emotions of the speaker, either for personal reflection or to increase the bandwidth of communication between people. Complex affective systems can be built to recognize and respond to the emotion of the speaker. This classical formation refers to the immediate nature of emotion experienced by human beings and attributed in some cultures and ways of thinking to all living organisms, and by scientific community to any creature that exhibits complex response traits similar to what human beings refer to as **Affect**. Affect is complex, and the term has no single universally accepted definition. Affects are mental states that arise spontaneously, rather than through conscious effort[1].

Affect as the subject of scientific research has multiple dimensions: behavioral, physiological, subjective and cognitive.

Recognition of affect is a complex task that is further more complicated by the fact that there is no unambiguous answer to what the correct affect is for a given speech sample. The difficulty with spontaneous affects is in their labeling, as the actual affect of the speaker is almost impossible to capture with certainty. Affects occurring in spontaneous speech are more difficult when compared with acted speech. In spontaneous speech, the occurrence of canonical affects such as happiness and anger is typically low. The distribution of classes is highly imbalanced making it difficult to measure and compare performance reported by different research people. Detecting affect in speech can be viewed as a classification task. It consists of assigning, an affect category (out of a fixed set), to a speech utterance. Accurate detection of affect from speech has clear benefit for the design of more natural speech interfaces.

1.1 Affect Scenario
Imagine you are called for an official meeting and members of the meeting other than you have not assembled in the meeting room. You have been looking forward to this meeting and have arrived five minutes early in anticipation. After fifteen minutes you feel quite anxious, and wonder if you are in the right meeting room and cross check the place and time. For the next few minutes you wait patiently, but after ten more minutes a range of feelings pass: concern, disappointment, frustration and then anger. After ten more minutes when all the members arrive, you try to read the situation or catch a hint of what may have caused the delay from their speech. You start listening empathetically to what others speak. When you speak a lot of gasps and voice tremor can be felt. Your speech paves a way to the flow of your emotions. Each person's tone and affect associated with the speech carries a greater importance than the message being conveyed. Conversations between members are minutely observed to bring a conclusion about their thoughts.

The above scenario illustrates how verbal cues play a vital role in social communication.

Affect recognition in speech would be a great achievement of computer technology. Affect computing is detecting the affect - psychological state of the user. Affect recognition classifies speech into categories of basic emotions for example angry, sad, etc,. The basic emotions, is widely used without implying that these emotions cannot be mixed to produce others. Identifying the affect of the individual is not as easy as it sounds. A comprehensible environment is taken for the research of affect.

In any organization, 80% of a higher official's time is spent in meetings. Meetings in an organization play a pivotal role in the growth and curtailment of the organization, since the decisions that are taken are directly proportional to meetings. If meetings go wrong, decisions made out of them are also erroneous and subsequently result in failures. Hence, it is very important to keep track of the success rates of meetings. There is no hard and fast rule that only one person should speak in the meeting. When more than one person speaks, the listeners tend to lose their attention and the required information is not conveyed by the speakers thus deviating from the goal. Now-a-days, all are open-meetings where each participant of the meeting has the freedom to talk and intervene at any point of time. When all the participants of the meeting want to talk and convey their mind, it leads to the problem of cross-talk and overlap in speech.

1.2 Emotion Detection

A widely accepted prediction is that computing will move to the background, weaving itself into the fabric of the everyday living spaces and projecting the human user into the foreground.

Consequently, the *future ubiquitous computing* environments will need to have human-centered designs instead of computer-centered designs. A change in the affective state of the user is a fundamental component of human-human communication.

Automatic recognition of emotion is gaining attention due to the widespread applications in various domains, including those with animated conversational agents. Automated recognition of emotion with high accuracy still remains an elusive goal due to the lack of complete understanding and agreement of emotion in human minds.

2. RELATED WORK

Recently, the recognition of emotions in speech has been extensively researched and various methods have been used. For example, Yu et al. [2] applied a multilevel structure based on coupled hidden markov models to estimate engagement levels in continuous natural speech. The continuous speech signal is segmented into spoken utterances and the acoustic features are computed from each utterance portion. The extracted non-linguistic information is used for predicting the emotional states such as discrete emotion types or arousal/valence levels by employing SVM-based classifiers. The HMM uses the previous information to model the emotional state of the user and engagement in conversation as a dynamic, continuous process.

Chateau et al. [3] presents a study of perception, the analysis and the modeling of styles or the *emotional quality* of speech. The speech emotional quality is evaluated in terms of the emotional content that describes the global impressions of the listener, as elicited by their audition. Specific subject criteria for evaluating the emotional quality are used to generate perceptive portraits of the speech. The evaluation is carried out by using linear models to connect the perceptive portraits to physical data derived from signal analysis. Some work has also been focused on using additional information regarding speech. The paper of [4] uses three sources of information, namely acoustic, lexical and discourse for recognizing emotions.

The work of [5] analyzes the strengths and the limitations of the systems based on the fusion of facial expression and acoustical information analysis at the decision level and about feature level integration. Kwon et al. [6] provides a comparison on the emotion recognition performance of various classifiers. SVM and HMM based classifiers produce significantly better results on SUSAS database from the previous approaches. A recent research of Rothkrantz et al. [7] focuses on studying the effect of the workload of speech production by making use of psychological experimental setup.

3. MATERIALS AND METHODS

The acoustic of speech can be correlated to the affective states. Pitch, energy, and speaking rate are widely observed to carry the most significant characteristics of affect in speech [8]. Stressed speech has an increase in mean pitch and mean intensity. Downward slopes are noted on pitch contour and there are increases in high frequency energy. Increased pitch range is also apparent. Neutral speech is shown to have a flatter pitch contour with energy more equally dispersed in the spectrum. Changes in pitch for stressed speech is said to be abrupt on stressed syllables and speaking rate is typically faster than neutral speech [9]. When compared to neutral speech, stressed sample has an increased pitch range and mean with the contour having abrupt downward slopes. The energy is much higher and is represented in the higher frequencies of the spectrum. The flow of the paper is depicted in figure 1.

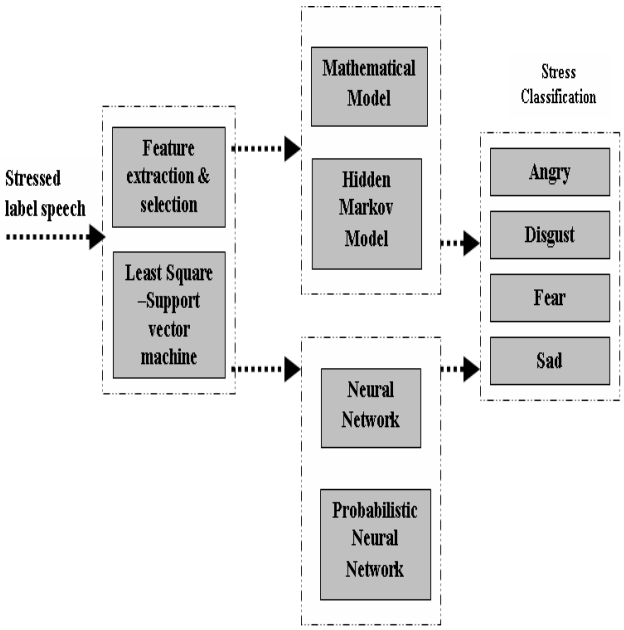

Figure 1. Flow diagram

3.1 Feature Derivation

Any pattern classification problem has to start with the first step of solution which is feature extraction. The method uses a least square support vector machine which computes sixty features from the stressed input utterances. A combinatorial feature selection approach that combats the feature extractor is adopted. The features are fed into a five state Hidden Markov mathematical Model(HMM) and a Probabilistic Neural Network(PNN). Both the methods classify the stressed speech into four basic categories of angry, disgust, fear, and sad. The PNN has superseded the mathematical model. The novelty in the technique has proved its superiority, which is highlighted in the results. Sixty Features were extracted from each stressed utterance, are listed in the table 1.

Table 1. Features calculated and their values

FEATURES	VALUES
Fundamental frequency(f_o)	Max, Min, Mean, SD, Mean and SD of f_o in max region, Mean and SD of f_o in min region, Positive slope in f_o- max, min, mean, SD, Negative slope in f_o- max, min, mean, SD, voiced-unvoiced ratio, Regression coefficient and its MSE, Value at first voiced segment, Value at last voiced segment
MFCC	20 Mel frequency cepstral coefficients
Energy	Mean, Min, Max, SD, Value at first voiced segment, Value at last voiced segment
Rhythm	Speaking rate, Avg. length of unvoiced segments (pause), Avg. length of voiced segments.

With the exception of those relating to rhythm, all features were calculated over the voiced segments of the sample [10]. A frame is flagged as unvoiced if it has no value for the fundamental frequency.

3.1.1 Feature Selection

It is well known that the presence of many irrelevant features may reduce the accuracy of classifiers. Feature selection is typically used to achieve three objectives:

- To reduce the size of the feature set in order to improve the prediction performance of the predictor
- To provide a fast and more computationally efficient predictor
- To provide a better understanding of the underlying process that generated the data.

3.1.2 Feature Selection Algorithm

A new feature selection algorithm has been proposed by [11]. It is based on a new filter like evaluation criterion called the Least Square Bound (LSBOUND) measure and has the advantage of both filter and wrapper methods. A criterion for feature selection is derived from the leave-one-out cross validation (LOOCV) procedure of the Least Squares Support Vector Machines (LS-SVM). It is closely related to an upper bound for LOOCV classification results. When an LS-SVM classifier is trained on the entire training set, if the corresponding Lagrangian multiplier α_p^o of the training sample x_p is positive, the following inequality holds in the leave-one-out procedure.

$$-y_p f^p(x_p) \le \alpha_o^p [(D^p_{\min})^2 + \frac{2}{\gamma}] - 1 \quad (1)$$

where α_p^o is the corresponding Lagrangian multiplier of x_p. When the LS-SVM classifier is trained on the entire training set, D^p_{min} is the distance between x_p and its nearest neighbor and γ is a positive value which penalizes errors. $f^p x_p$ is the validation result for the sample x_p in the leave-one-out procedure. If $y_p f^p(x_p)$ is negative, the sample x_p is considered as a leave-one-out error, and if $y_p f^p(x_p)$ is positive, x_p is correctly classified in the leave-one-out procedure.

Implementation

The feature selection task in the emotion detection problem is performed as follows.

In the **first stage**, the feature selection in the multi-class emotion detection problem is put into a form of a binary classification problem. First, the emotion detection problem has been treated in a *one-vs-rest* framework. In this framework, the classification problem is to discriminate one specific class of emotion from the rest. Class specific features, which are important in terms of distinguishing one class from the rest, are expected to be selected by the feature selection algorithm. As there are M classes of emotional states, M subsets of features are selected by each of the feature selection algorithms.

In the **Second stage**, which is called the feature construction stage, the subsets S_i, produced by the feature selection algorithm in the first stage are processed to finally obtain the *best feature subset*. In this stage, two strategies are used to construct a final feature set. First, features that occur more than once in any of the subsets S_i are added into the set to form the *best final subset*. This final subset of features is labeled as SET1-INT and it will simply be the intersection of the subsets of features S_i given by

$$INT(S_i) = \bigcup_{i,j=1,i\neq j}^{N} S_i \bigcap S_j \quad (2)$$

where N=M is equal to the total number of subsets S_i.

In the second strategy, the subsets of features S_i are simply combined together to obtain the final subset, which is labeled SET1-UNI. This task corresponds basically to performing a unification operation on the subsets S_i as follows:

$$UNI(S_i) = \bigcup_{i=1}^{N} S_i \quad (3)$$

3.1.3 Cross Validation Error Calculation

Select each training example in turn as the single example to be held out, train the classifier on the basis of all the remaining training examples, test the resulting classifier on the *one* held out and count the errors.

Classifier form:

$$f(X, \theta, \theta_o) = sign(\theta^T X + \theta_o) \quad (4)$$

where, θ -normal to the separating hyperplane, θ_o-offset parameter, N=M equals the total number of subsets S_i.

Table 2. CV Error calculation

LSBOUND	NO. OF FEATURES	CV ERROR
SET1_INT	39	0.117
SET1_UNI	41	0.187
Average	40	0.1505

The average CV error produced by the framework for feature selection is stated in table 2. The number of features derived by equation (2) is 39 and by (3) is 41. From this an average of 40 features are accounted with average CV error of 0.1505. The feature selection has taken 40 out of the 60 features.

3.2 Mathematical Model

HMMs have been widely used for many modeling and classification problems [12]. The most common application of HMM is in speech recognition. One of the main advantages of HMMs is their ability to model non stationary signals to events. This signal is non stationary in nature, since an expression can be given in varying rates with varying intensities even for the same individual.

A HMM is given by the following set of parameters

$$\lambda = (A, B, \prod) \quad (4)$$

$$a_{ij} = P(q_{t+1} = S_j \mid q_t = S_i) 1 \le i, j \le N \quad (5)$$

$$B = \{b_j(O_t)\} = P(O_t \mid q_t = S_j) 1 \le j \le N \quad (6)$$

$$\prod_j = P(q_1 = s_j) \quad (7)$$

where A is the state transition probability matrix, B is the observation probability distribution and Π is the initial state

distribution. The number of states of the HMM is given by N. It should be noted that the observations O_t can be either discrete or continuous. Given a HMM, there are three basic problems that are of interest.

- Given the model how to efficiently compute the probability of the observations, given the model. This problem is related to classification in the sense that it gives a measure of how well a certain model describes an observation sequence.
- Given the set of observations and the model, to find the corresponding state sequence in some optimal way.
- Given the set of observations and the model how to learn the parameters of the model λ, it is necessary to maximize the probability of observation of the given speech sequence.

3.2.1 Architecture

There will be four HMMs one for each expression: {angry(1), disgust(2), fear(3) and sad(4)}. The observation vector O_t, for the HMM represents continuous motion of the units. Therefore B is represented by the probability density functions (pdf) of the observation vector at time t given the state of the model. The Gaussian distribution is chosen to represent these pdf's i.e.,

$$B = \{b_i(O_t)\} \approx \mathrm{N}(\mu_j, \Sigma_j), 1 \le j \le N \qquad (8)$$

where μ and Σ are the mean vector and full covariance matrix respectively. The parameters of the model of expression or specific HMM are learnt using the Baum-Welch re-estimation formulas. The Baum algorithm is used to derive the maximum likelihood (ML) estimation of the model parameters [13].

Parameter learning is followed by the construction of the Maximum Likelihood (ML) classifier. Given an observation sequence O_t where t Є (1,T), the probability of the observation, given to each of the four models $P(O_t|\lambda_j)$, is computed using the forward backward procedure. The sequence is classified as the emotion corresponding to the model that yielded the highest probability ie.

$$C^* = \arg\max 1 \le c \le 5[P(O \mid \lambda_c)] \qquad (9)$$

The main problem with the approach taken is that it works on isolated speech expression sequences or on pre-segmented sequences of the expressions. In reality, this segmentation is not available and therefore there is a need to find an automatic way of segmenting the sequences. Concatenating of HMMs representing the phonemes in conjunction with the grammar has been used for continuous speech recognition system.

3.2.2 Automatic segmentation and recognition of emotions using multilevel HMM

The figure 2 shows the architecture for HMM. The features are fed continuously to the four emotion specific HMMs. The state sequence of each of the HMMs is decoded and used as the observation vector for the high level HMM. The high level HMM consists of five states, one for each of the four emotions and one for neutral. Whenever there is no emotion in the speaker's speech for a longer duration it is necessary to deal it as neutral state. The transitions between emotions are imposed through the neutral state since it is fair to assume that the voice resumes a neutral position before it exhibits a new emotion. For instance, a person cannot go from expressing happy to sad without returning the voice to its neutral position.

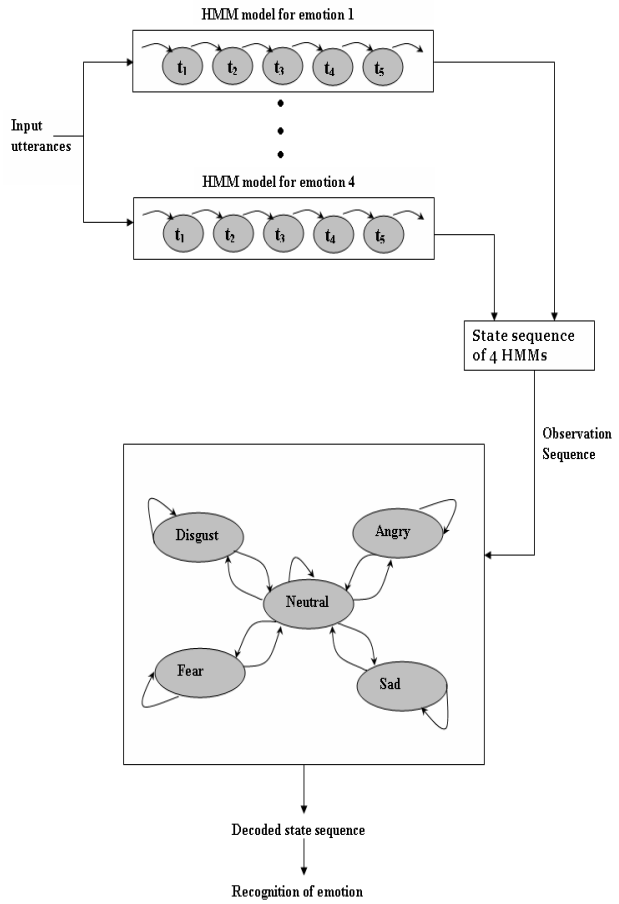

Figure 2. Multilevel HMM Architecture

The recognition of the expression is done by decoding the state that the high level HMM is in at each point of time, since the state represents the displayed emotion. To get a more stable recognition output of the classifier, it will actually be a smoothed version of the state sequence i.e., the high level HMM will have to stay in a particular state for long enough in order for the output to be the emotion related to that state.

The training procedure of the system is as follows:

MULTI-LEVEL HMM ALGORITHM

Step 1: Train the emotion specific HMMs using a manually segmented sequence.

Step 2: Feed all four HMMs with the continuous (labeled) speech sequence. Each speech sequence may contain several instances of each emotion with neutral instances separating the emotions.

Step 3: Obtain the state sequence of each HMM to form the four dimensional observation vector of the higher level HMM, i.e., $O_h(t) = [q_t^{(1)},\ldots,q_t^{(4)}]^T$ where $q_t^{(i)}$ is the state of the i^{th} emotion specific HMM. The decoding of the state sequence is done using the viterbi algorithm.

Step 4: Learn the probability observation matrix for each state of the high level HMM using $P(q_j^{(i)}|S_k)$ = {expected frequency of model i, being in state j, given that the true state was k} and

$$B^{(h)} = \{b_k(O_t^h)\} = \{\prod_{i=1}^{4}\prod(P(q_j^{(i)} \mid S_k)\} \quad (10)$$

where j Є (1, Number of states for lower level HMM).

Step 5: Compute the transition probability A= $\{a_{kl}\}$ of the high level HMM using the frequency of transiting from each of the four emotion classes to the neutral state in the training sequences and from the neutral state to the other emotion states. For notation, the neutral state is numbered 5 and other states are numbered as 1 to 4 from angry, disgust, fear to sad. It should be noted that the transition probabilities from one emotion state to another that is not neutral are set to zero.

Step 6: Set the initial probability of the high level HMM to be 1 for the neutral state and 0 for all the other states. This forces the model to always start at a neutral state and assumes that a person will display a neutral expression in the beginning of any utterance. This assumption is made for the simplicity of the testing.

The steps followed during the testing state are the same as followed during training. The stress labeled speaker tracking sequences are fed into the lower level HMMs and a decoded state sequence is obtained using the viterbi algorithm. The decoded lower level state sequence O_t^n s fed into the higher level HMM and the observation probabilities are computed using the equation (10). It is assumed that the state sequences of the lower level HMMs are independent, given the true labeling of the sequence. This assumption is reasonable since the HMMs are trained independently and on different training sequences. In addition, without this assumption, the size of B will be enormous, since it will have to account for all possible combinations of the states of the four lower level HMMs and it would require a huge amount of training data.

Using the Viterbi algorithm again for the high level HMM, a most likely state sequence is produced. The state that the HMM was in at time t corresponds to the expressed emotion in the sequence at time t. To make the classification result robust, to understand fast changes, a smoothing of the state sequence is done by not changing the actual classification result. If the HMM did not stay in a particular state for more than T times, where T can vary between 1 and 15 samples, a smoothing is applied. The introduction of the smoothing factor T will cause a delay in the decision of the system but of not more than T sample times.

3.3 Neural Network Classifier

ANNs have gained prominence in the area of pattern recognition, and have several properties that make them attractive for speech. These include

- Simple implementation.
- Parallel algorithm.
- Robustness to noise.
- Self-learning ability.

3.3.1 Architecture

The Probabilistic Neural Network (PNN) introduced by Donald Specht in 1988, is a three layer feed forward one-pass training algorithm used for classification and data mapping. It is based on well-established statistical properties derived from Baye's theorem and non-parametric kernel based estimators of probability density function. The class probability density functions are smooth and continuous which is a major advantage of PNN.

The network contains an input layer which has as many elements as there are separable parameters needed to describe the objects to be classified [14]. The network architecture is depicted in figure 3.

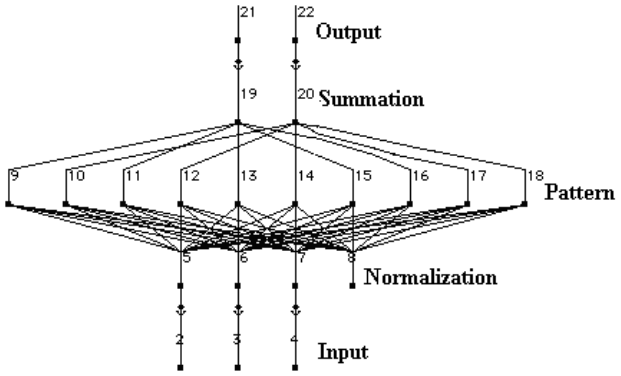

Figure 3. PNN Architecture

The network consists of 4 layers namely input, pattern, summation and output layer. The feature vectors are provided in the input layer. It has a pattern layer, which organizes the training set such that each input vector is represented by an individual processing output layer called the summation layer, which has as many processing elements as there are classes to be recognized. Each element in summation layer combines via processing elements within the pattern layer which relate to the same class and prepares that category for output.

There is an additional layer added to normalize the input vector. The normalization process done here subtracts the median and divides the inter-quartile range. In the pattern layer, there is a processing element for each input vector in the training set. Each processing element in the pattern layer is trained once. An element is trained to generate a high output value when an input vector matches the training vector. In any case, the training vector need not be in a particular order. The learning function selects the first untrained processing element in the correct output class and modifies its weight to match the training vector. The pattern layer operates competitively where only the highest match to an input vector wins and generates an output.

The PNN is a classifier that possesses good generalization properties and more importantly the time required for designing the network is less. The design of the network is straightforward and does not depend on training. The best feature with PNN is with less training data, fast and consistent training can be incorporated.

3.3.2 Implementation

The PNN implements the parzen window estimator by using a mixture of Gaussian basis functions. If a PNN for classification in k classes is considered, the probability density function $f_i(x_p)$ of each class k_i is defined by the equation

$$f_i(x_p)=\frac{1}{(2\Pi)^{\frac{d}{2}}\sigma^d}\frac{1}{M_i}\sum_{j=1}^{M_i}\exp\left(\frac{1}{2\sigma^2}(x_p-x_{ij})^T(x_p-x_{ij})\right)\text{wh}$$

ere i = 1,2,..., k, x_{ij} is the j^{th} training vector from class k_i; x_p is the p^{th} input vector, d is the dimension of the feature vectors and M_i is the number of training patterns in class k_i. K here takes the value of 5 corresponding to angry, disgust, fear, happy and sad. Happy affect is considered here for a basic reason that naming a meeting as positive implies that all the sub scores were positive on the meeting. A margin of error would have come in as, if one of the sub scores were positive. To give consideration to this aspect, a class of Happy is again taken. Each training vector x_{ij} is assumed to be a center of a kernel function and consequently the number of pattern units in the first hidden layer of the NN is given as the sum of the pattern units for all the classes [15]. The variance σ acts as a smoothing factor which softens the surface defined by the multiple Gaussian functions. σ has the small value for all the pattern units. Therefore, a homoscedastic PNN is considered. The Bayesian decision rule is applied to distinguish class k_i to which input vector x_p belongs:

$$D(x_p)=\arg\max_i\{h_iC_if_i(x_p)\}\quad \text{i=1,2,...,K} \qquad (11)$$

where h_i is a-priori probability of occurrence of the patterns of category k_i and c_i is the cost function in case of misclassification of a vector belonging to class k_i.

4. RESULTS AND ANALYSIS

The ICSI Meeting Recorder Database[ ] was considered for the experiments. The database consists of 75 meetings in total out of which 63 was taken for the experiment. The 63 meetings were labeled as “Stress affect meetings”. The meeting transcript files were available in 8 names: BMR, Bro, Bed, Bns, Buw, Bsr, Bdb and Btr. The results obtained by using the novel (PNN) method and HMM (for the 63 meetings) for the ICSI meeting recorder database is tabulated in table 3.

Table 3. Accuracy classification for the methods adopted in the work.

CATEGORIES	HMM (%)	PNN (%)
Angry	92.87	99.55
Disgust	91.97	99.88
Fear	89.88	97.86

Happy	88.79	97.35
Sad	90.23	92.02
Average	**90.7**	**97.1**

It is very clear from the table, that the suggested approach has outperformed the HMM model with regard to neutral and stressed speaking styles. Moreover, the intended approach has produced an emotive score of 97.1% when compared with the accuracy of 90.7% generated by the HMM model underlying its supremacy.

Table 4. Classification for different meetings.

MEETING NAME	ANGRY	DISGUST	FEAR	HAPPY	SAD
BMR	**8**	**3**	**4**	**3**	**5**
Bro	**8**	**2**	**6**	**1**	**4**
Bed	**5**	**1**	**3**	**1**	**2**
Bns	-	**1**	-	-	**2**
Btr	-	**2**	-	-	-
Bdb	-	-	**1**	-	-
Buw	-	**1**	-	-	-
Bsr	-	-	-	-	-
Total	21	10	14	5	13

The table 4 gives the classification rate considering the holistic input of the 63 stress or negative labeled meetings. 21 meetings are cataloged as *Anger affect*, 10 meetings under *Disgust affect*, 14 under *Fear* label, 5 with *Happy*, and 13 as *Sad*. A proper justification can be made in happy category as all the utterances given for the network were stressed category and the major of happy affect has already been distilled in the meetings classification based on the sub-scores. The Bsr meeting did not reveal any entries in the table as, the set contained only one meeting and it was classified as a positive meeting. There is no entry in the negative classification.

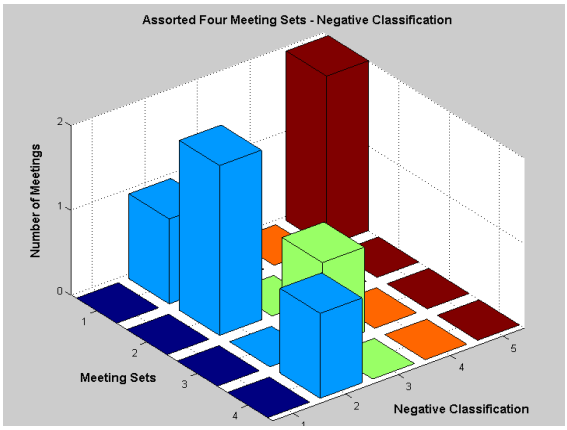

Figure 4. Assorted meetings set classification results

Five meetings were considered assorted from the set of 63 meetings. They are Bns, Btr, Bdb, Buw, and Bsr. The Bsr is not taken into account here since it is already has been classified into a Positive Meeting. The assorted set here refers to the four meeting sets. The figure 4 has lucidly plotted the classification results for the assorted set. The z axis specifies the number of meetings, y axis the meeting sets considered i.e., 4, and in x axis the classification was made into five classes. The y axis meeting set pertains to the corresponding assorted meetings as Bns, Btr, Bdb, and Buw. The x axis designates the negative classes - anger, disgust, fear, happy and sad. Each class in the meeting set 1 ie., Bns Meeting set has made a categorization of 0 meetings in anger class, 1 in disgust, 0 in fear and happy and 2 in the sad category. The meeting set 2 Btr is categorized 0 meeting in anger, 2 in the disgust category, 0 in the fear, happy and sad categories. The third meeting set Bdb has been sorted out into 1 meeting in fear affect category and 0 in all the other classifications. The final meeting set in the assorted framework Buw has one meeting which is in the disgust category.

Figure 5 symbolizes the BMR meeting set negative classification. 8 meetings were labeled as anger, 3 under disgust, 4 fear category, 3 as happy and 5 meetings into sad. The x axis display the classification of BMR meeting set, y axis represents Negative classification pertaining to angry, disgust, fear, happy and sad. The z axis points to the number of meetings. The total sets of meetings were 23 negative meetings in the BMR set.

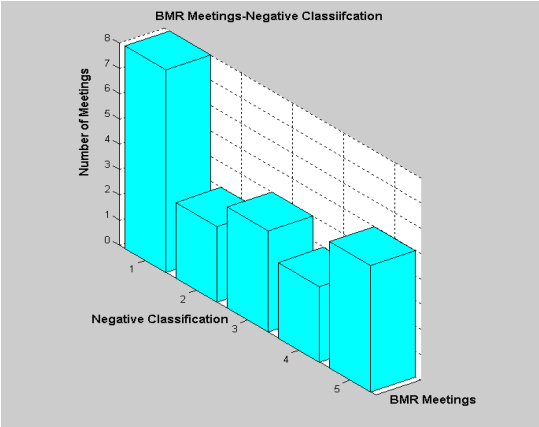

Figure 5. Graphical representation of classification for BMR meeting Set

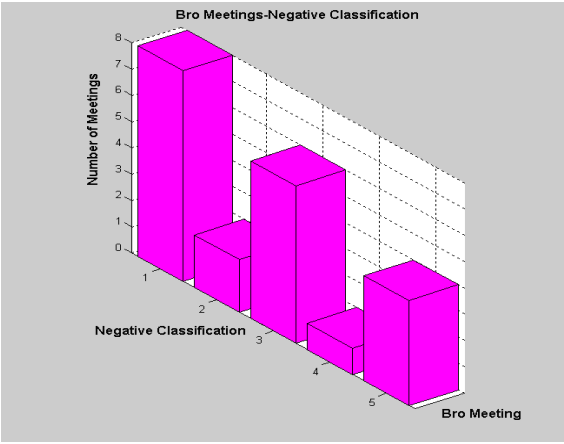

Figure 6. Graphical representation of classification for Bro meeting Set

The figure 6 depicts the Bro meeting set classification results. The Bro meeting set comprised 23 meetings under negative catalog. The x axis in the 3D bar, display the classification of Bro meeting set, y axis represents negative classification pertaining to angry, disgust, fear, happy and sad. The z axis points to the number of meetings. 8, 2, 6, 1, 4 are the number of meeting sets pertaining to the angry, disgust, fear, happy and sad labels of the Bro meeting set.

Figure 7 illustrates the Bed meeting set results. The axis denotes the Bed meeting sets on one hand and the Negative classification on the other. There are a total of 12 Bed meetings, which are partitioned as 5 under anger, 1 disgust meeting, 3 fear affects, 1 happy category and 2 with the sad label. A maximum of the 5 are under the first category of angry.

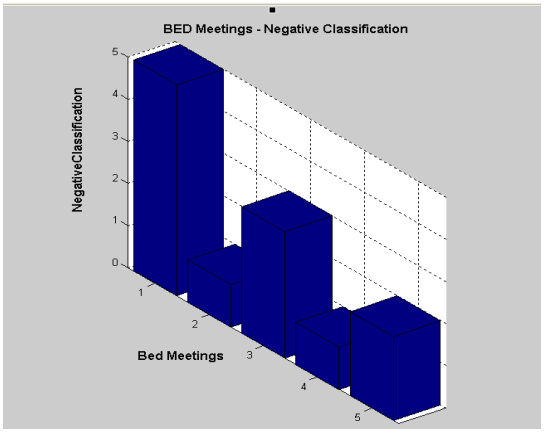

Figure 7. Graphical representation of classification for BED meeting Set

5. CONCLUSION & FUTURE WORK

The detailed structures of speech, that contributes to the affect model and their interactions at fairly abstract levels were analyzed in this study. To corroborate this, the system's final score on the result of the meeting has proved to be concrete. The work gives a lucid conclusion of the five class classification. From the 63 negative meetings, 33% were classified as Angry class, 16% as Disgust affect, Fear had 22% of meetings, the Sad affect 21% and finally the Happy, which is not always considered as stressed class had 8% of the meetings.

Since, Affect recognition is not an exact science but an approach based on user-defined probabilities, varying the user-defined threshold can change the performance. Discrete labeling of affect varies from one person to another in the field of affect computing making it difficult to work in this arena. Also, by inducing manual alignment the performance is found to augment

6. REFERENCES

[1] R Cowie, E Douglas, N Tsapastsoulis, G Votis, S Kollias, W Fellenz, J G Taylor, “Emotion Recognition in Human Computer Interaction”, *IEEE Signal Processing Magazine*, 18, (2001), 32-80.

[2] Doddington G, “Speaker Recognition based on Idiolectal differences between Speakers”, *In Proc. European Conference on Speech Communication and Technology*, (2001), 2521-2524.

[3] Chateau N, Maffiolo V, Ehrette T, Alessandro C, “Modelling the Emotional Quality of Speech in a Telecommunication context”, *In Proc. of the International Conference on Auditory display*, (2002), 269-274.

[4] Lee C M, Narayanan S S, "Toward Detecting Emotions in Spoken Dialogs", *In Proc. of IEEE transactions on Speech and Audio Processing*, 13(2), (2005), 293-303.

[5] Busso C, Deng Z, Yildirim S, Bulut M, Lee C M, Kazemzadeh A, Lee S, Neumann U, Narayanan S, "Analysis of Emotion Recognition using Facial expression, speech and Multimodal interfaces, *ACM Press*, (2004), 205-211.

[6] Kwon, Oh-Wook, Chan K, Hao j, Lee, Te Won, "Emotion Recognition by Speech Signals", *In Proc. of Eurospeech,* (2003), 125-128.

[7] Rothkrantz L J M, Wiggers P, Van Wees J W A, Van Vark R J, "Voice Stress Analysis", *In Proc. of Text, Speech and Dialogues*, (2004), 449-456.

[8] Huber R, Batliner A, Buckow J, North E, Warnke V, Neihmann H, "Recognition of Emotion in a realistic Dialogue System Scenario", *In proc. of IEEE international Conference on Spoken Language Processing*, 1, (2000), 665-668.

[9] Donn Morrison, Ruili Wang, Liyanange C De Silva, "Spoken Affect classification using Neural Networks", *In Proc. of IEEE International Conference on Granular Computing*, 2, (2005), 583-586.

[10] Andrel Mihalia, "Advanced Digital Signal processing", *Speech Processing Lecture*, 12, (2004).

[11] Altun H, Shawe Taylor J, Polar G, "New Feature Selection Frameworks in Emotion Recognition to evaluate the Informative power of Speech related Features", *In Proc. of IEEE International Conference of Information science, Signal Processing and their Applications*, (2007), 1-4.

[12] Ira Cohen, Ashutosh Garg, Thomas S Huang, "Emotion Recognition from Facial expressions using Multi level HMM", *In Neural Information Processing systems workshop on Affective Computing*, (2000).

[13] Levinson S E, Rabiner L R, Sondhi M M, "An Introduction to Application of the Theory of Probabilistic Functions of a Markov process to Automatic Speech Recognition", *In Bell Lab System Technical Journal*, 62(4), (1983), 1035-1072.

[14] Raymond Low, Roberto Togneri, "Speech recognition using the Probabilistic Neural Networks", In Proc. of IEEE *International Conference on Spoken Language Processing*, (1988).

[15] Panagiotis Zervas, Todor Ganchev, Nikos Fakotakis, "Negative Emotional State detection from Speech", *In symposium on Communication systems, Networks and Digital Signal Processing*, (2006), 310-313.