RANDOM DISCRIMINATIVE PROJECTION-BASED FEATURE SELECTION FOR COMPUTATIONAL PARALINGUISTICS

by

H. Tuğçe ÖZKAPTAN

B.S, Computer Engineering, Dokuz Eylul University, 2011

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Computer Engineering

Boğaziçi University

2014

RANDOM DISCRIMINATIVE PROJECTION-BASED FEATURE SELECTION FOR COMPUTATIONAL PARALINGUISTICS

APPROVED BY:

Prof. S. Fikret Gürgen (Thesis Supervisor)	.....
Assist. Prof. Arzucan Özgür	.....
Assoc. Prof. Olcay Kurşun	.....

DATE OF APPROVAL: 08.08.2014

ACKNOWLEDGEMENTS

Foremost, I would like to express my sincere gratitude to my thesis supervisor Prof. S. Fikret Gürgen for his patience, encouraging comments, guidance and measurable support. Besides my advisor, I would like to thank the rest of my thesis committee members, Assist. Prof. Arzucan Özgür and Assoc. Prof. Olcay Kurşun for their discussion on this work and Heysem Kaya who developed initiative work of my thesis, motivated me to enhance his work and gave his all technical and mental support during my study. His guidance helped me in all the time of research and writing of this thesis.

My sincere thanks also goes to thank TUBITAK for keeping their financial support on me during my graduate education.

I would want to thank to my fiancé, Ali Yavuz Kahveci, who encouraged me, fed me and made me feel I am not alone. I would also want to thank my lovely brother, my home mate, Buğra Kaan Ozkaptan and my father, Cihat Ozkaptan for keeping their support on me all the time and their encouraging talks during my long hardworking hours.

Up to this stage of my life, I have always felt the support and trust of my mother in any work that I am included and in any activity that I perform. Therefore, I would specially like to thank Berrin Ozkaptan, I would like to dedicate my thesis to her.

ABSTRACT

RANDOM DISCRIMINATIVE PROJECTION-BASED FEATURE SELECTION FOR COMPUTATIONAL PARALINGUISTICS

Computational paralinguistics deals with the underlying meaning of the verbal messages. Understanding the meaning of verbal messages provides interpreting spoken content and behaving accordingly like humans. It allows us to develop human like machines. Hence, paralinguistic area is attracting increasing attention for research. Paralinguistic analysis involves extracting features from raw speech data, chunking, selecting relevant features and training the model. In this thesis, the focus is on the feature selection step. Feature selection aims at finding a relevant and necessary set of features to train generalizable models. The main challenge for feature selection methods is the greedy-search nature of them. One major motivation for this study to develop an efficient feature selection technique is the success of a recently developed discriminative projection based feature selection method. Here, the method is enhanced by applying the power of stochasticity to overcome traps in local minimum while reducing the computational complexity. The proposed approach assigns weights both to groups and to features individually in many randomly selected contexts and then combines them for a final ranking. The efficacy of the proposed method is shown in two recent challenge corpora to detect level of depression severity and conflict.

ÖZET

HESAPLAMASAL PARALİNGUİSTİK İÇİN RASSAL AYRIMSAYICI İZDÜŞÜM TABANLI ÖZNİTELİK SEÇİMİ

Hesaplamasal Paralinguistik, sözlü mesajın gerçek anlamıyla ilgilenir. Sözlü mesajları anlamak insanlar gibi onları yorumlayıp buna göre davranabilme imkanı verir. Bu da bize insan benzeri makineler geliştirebilme olanağı sağlar. Bu yüzden, paraliguistik alanı araştırma için giderek artan seviyede alaka topluyor. Paralinguistik analiz işlem görmemiş veriden öznitelik çıkarımı, yığınlama, öznitelik seçimi ve model eğitimi adımlarını kapsar. Bu tezde odaklanılan adım öznitelik seçimidir. Öznitelik seçimi genellenebilir model eğitebilmek için ilgili ve gerekli özniteliklerin bulunmasını amaçlar. Öznitelik seçimi yöntemlerindeki temel sorun onların fırsatçı algoritmaya dayalı arama tabiatıdır. Bu çalışmadaki başlıca motivasyonumuz yakın zamanda geliştirilmiş olan ayrımsayıcı izdüşüm tabanlı öznitelik seçimi yönteminin başarısıdır. Burada bu yöntem, yerel mimimumlarda takılmanın üstesinden gelmek ve hesaplama karmaşıklığını azaltmak için rassallığın gücüne başvurularak geliştirilmiştir. Önerilen yöntem öznitelik gruplarını ve özniteliklerin kendilerini bir çok rassal seçimli ortamda ağırlıklandırıp sonrasında nihai sıralama için birleştirir. Önerilen yöntemin etkinliği depresyon ciddiyet seviyesini ve çatışma seviyesini tespit amacıyla yakın zamanda düzenlenen iki müsabaka probleminde gösterilmiştir.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

A	Affinity matrix
$\overline{\mathcal{A}}$	Localized Discriminative Affinity matrix
C_{XY}	Cross-set Covariance
C_{XX}	Within-set Covariance
S^w	Within class Covariance
S^b	Between class Covariance
T	Target Variable Matrix
U_X	Covariates obtained from dataset X
V, W	Projection Matrices
X	Dataset
ρ	Canonical Correlation Value
λ	Eigen value

LIST OF ACRONYMS/ABBREVIATIONS

ARD	Automatic Relevance Determination
AVEC	Audio-Visual Emotion Corpus and Challenge
A/D Converter	Analog-to-Digital Converter
CCA	Canonical Correlation Analysis
CFS	Correlation Based Feature Selection
CSC	Conflict Sub-challenge
DCCA	Deep Canonical Correlation Analysis
DSC	Depression Sub-challenge
F0	Fundamental Frequency
FDA	Fisher Discriminant Analysis
GA	Genetic Algorithm
GP	Genetic Programming
GPFS	Genetic Programming Based Feature Selection
HMM	Hidden Markov Model
KCCA	Kernel Canonical Correlation Analysis
LFDA	Local Fisher Discriminant Analysis
LPP	Locality Preserving Projection
MAE	Mean Absolute Error
MFCC	Mel-Frequency Cepstral Coefficients
MI	Mutual Information
MRMR	Minimum Redundancy Maximum Relevance
PCA	Principal Component Analysis
PCC	Pearson's Correlation Coefficient
RMSE	Root Mean Square Error
SBE	Sequential Backward Elimination
SFS	Sequential Forward Selection
UAR	Unweighted Average Recall

1. INTRODUCTION

Paralinguistics means 'alongside linguistics' (from the Greek preposition $\pi\alpha\rho\alpha$) [1]. This discipline is interested in the meaning of verbal message in communication rather than spoken content. The verbal message can be in acoustic form (vocal, non-verbal phenomena) or in linguistics form (connotations of single units or of bunches of units). In daily life, we continuously apply verbal message consciously or un-consciously to explain what is in our minds, to show our feelings and to express ourselves in the light of communication.

People track their partner's age, gender, mood, emotion, intention and then re-adapt their manner of speaking accordingly in human-human communication. They also have the ability of interpret their partner's attention. Capability of inference and react like a human still an unachievable problem and paralinguistic is constantly developing into major field of speech analysis, as new human-machine interaction advance over sheer speech recognition.

Schuller *et al.* [1] bundle paralinguistics under the heading of *speaker classification* based on speech and language analysis. The authors point out to various scenarios where speaker classification could be applied. Some of these scenarios have been mentioned in the literature or in media repeatedly while some of them already deployed as a real-world application. A few examples for such scenarios are listed below to show how paralinguistic speech analysis results in great advance in human–machine interaction as well as in machine mediated human–human communication.

In the field of *multimedia retrieval*, paralinguistic information is of interest for manifold types of media searches, such as highlights in sports games by measuring the level of excitement in the reporter's speech or simply looking for speakers belonging to specific classes (such as age, gender, or charisma [2]). In *robotics*, the analysis of affective states (emotion, feeling) and personality is still very rudimentary in robotics and often limits itself to tactile interactions. With a better modeling of these states and

traits, we will be able to add social competence to humanoid or other highly interactive and communicative robots [3,4], assistive robots [5], or to (virtual) agents [6]. In *health area*, speech based classification can be used to help elderly people to live longer in their homes by using an acoustic pain surveillance for detecting and classifying distress calls automatically without using automated speech recognition [7]. On the other hand, voice classification can be used to diagnose, monitor, and screen diseases and speech disorders [8] such as Parkinson's disease [9], patients who had their larynx removed due to cancer, children with cleft lip and palate [10] or dysphonia [11]; or further pathological effects [12]. Motivated by the success of previous works in health domain, recent challenges introduce biomedical corpora e. g. for autism detection/diagnosis [13], and depression level prediction [14] in order to help advance the field by providing comparability and transparency to state-of-the-art studies.

As paralinguistic speech analysis includes many views, they can be gathered under three main headings: speaker state, speaker trait and vocal behaviour. Speaker state relates to the states changing over time i.e affection and intimacy [15], deception [16], emotion [17],interest [18], intoxication [19], sleepiness [20], health state [21], and stress [22] or zest. Speaker trait relates to permanent characteristics such as age and gender [1], height [23], likeability [24], or personality. And the vocal behavior relates with the non-verbal behavior such as sighs and yawns [25],laughs [26], [27], hesitations and consent [18], and coughs [28].

When the spoken content rather than the speaker characteristics are concerned, variants of Hidden Markov Models (HMM) are the industry standard. However, as long as the states and traits are concerned, not the instantaneous changes of acoustic features but their relative longer term (from few seconds to few minutes) summary is important, due to the supra-segmental nature of the underlying phenomena [29]. Thus, the state-of-the-art-results in the field are obtained by mapping the time-varying Low Level Descriptor (LLD) contours (e. g. F0, MFCC 1-12, jitter, shimmer) onto a scalar by means of a summarizing functional thus obtaining a fixed length vector. The applied functionals can range from moments (e. g. mean, variance, kurtosis) to extremes (e. g. min, max), and to coefficients of polynomials fit to these contours. This approach

is being used in the INTERSPEECH COMPARE challenge series as baseline since 2009 [30], where the authors present 384 suprasegmental features. Benefiting from the openSMILE toolkit that extracts a wide range features based on this approach [31], the dimensionality and quality features increased in time. In COMPARE 2013 challenge, the organizers provided a baseline acoustic feature set having a with 6 373 features. Despite the success of the brute-forcing approach, this very high-dimensional feature sets include a bulk of irrelevant and redundant features; and prone to *curse of dimensionality* as the number of samples is usually at the order of few hundreds.

Since the feature extraction is not a bottleneck in the state-of-the-art pipeline, which will be discussed in more detail in the next section, in this thesis reduction of brute-forced openSMILE features is on focus. Although there are many successful feature selection algorithms in literature, finding effective feature set in an efficient way is still a challenging problem. In a recent study, Kaya *et al.* [32] proposed the use of Canonical Correlation Analysis (CCA) to rank acoustic features for prediction of level of depression. Although this method outputs results better than baseline, even state-of-the-art at time of publication, it is still open to improvement. One avenue for improvement, as in our recent work [33], is application of the discriminative projections to domain information-based partitions of data in a divide-and-conquer manner. One other direction is using stochasticity instead of domain knowledge to form the groups.

In my thesis, the motivation and the primary research direction is to improve the method by applying a discriminative projection to randomly selected features, thus obtaining an ensemble. Due to its usability both in regression and classification tasks, CCA is used as discriminative projection. While preliminary work also included Local Fisher Discriminant Analysis (LFDA), which is only applicable to classification. The hypothesis is tested on very high dimensional recent challenge corpora: INTERSPEECH 2013 Computational Paralinguistics Challenge-Conflict sub-challenge (CSC) [13] and AVEC 2013 Challenge-Depression sub-challenge(DSC) [14]. In both corpora development set results obtained by the proposed method compare favorably to challenge baselines as well as benchmarking SLCCA-Filter method.

The remainder of this thesis is organized as follows. In Chapter 2, background information on speech processing pipeline and relevant methods is given. In Chapter 3 the proposed method is presented. Chapter 4 provides experimental results on the tested challenge corpora, whereas Chapter 5 concludes with future directions.

2. BACKGROUND

In this chapter, the background information about paralinguistic speech processing, feature reduction methods as well as related literature works is given.

2.1. Paralinguistic Speech Processing

Before going further in paralinguistic speech analysis, i. e. the way we can extract 'how' something is said rather than 'what' is said; it will be suitable to give basic definitions of voice and speech. Voice (or vocalization) is the sound produced by humans and other vertebrates using the lungs and the vocal folds in the larynx, or voice box. Speech is the verbal means of communicating and it is produced by precisely coordinated muscle actions in the head, neck, chest, and abdomen[1] . In the light of these general definitions, we refer the characteristic of speaker's voice as *voice*, while the spoken language adding linguistic as *speech* in paralinguistic view.

Initially, one should handle the computational intelligence analysis from the speech as a general pattern recognition paradigm. In pattern recognition, we learn a model with pre-processed training data and then test the performance of the learner with test data. The steps for constructing such a model are demonstrated in Figure 2.1.

Pre-processing step deals with extracting signal properties from raw speech data. This raw data may be stored speech file or A/D converter output. Pre-processing step also includes the removal of noise from speech signals. *Feature extraction* refers to converting speech information to acoustic and linguistic representations.

Speech database usually contains audio stored files with labels (i.e age, gender, emotion of speaker) of concerning job. Sometimes, spoken content with related speech record is also found in database.

[1]http://www.nidcd.nih.gov/health/voice/pages/Default.aspx

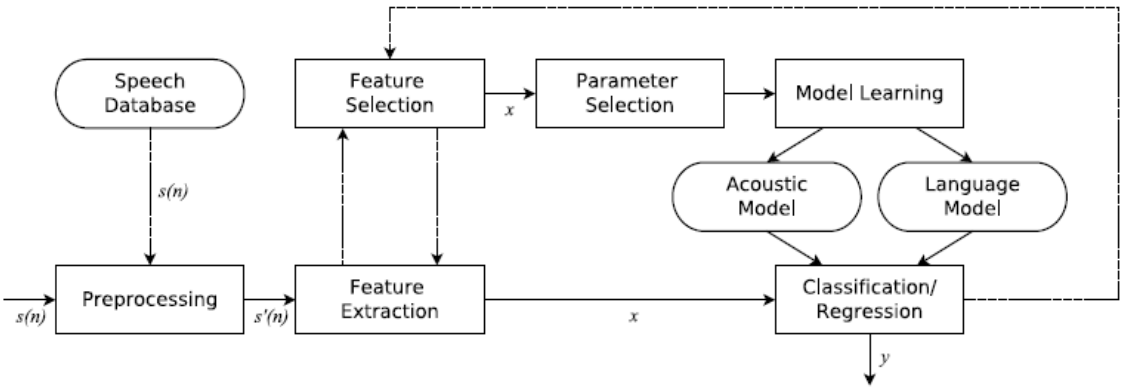

Figure 2.1. A unified model for The Computational Speech Analysis [29].

Model learning refers to training the classifier or regressor with the labeled data in speech database. *Parameter Selection* deals with the finding optimum values for parameters specific to learner model. *Acoustic and Language Model* are mainly resembles each other except that The Language Model also includes the dependency for linguistic information.

Classification/Regression takes place after the model learning. Classifier tries to predict discrete class labels or binary value like low/high. In case of regression, the output is a continuous value like age of speaker or dimensions like potency,arousal, and valence, typically ranging from -1 to $+1$. The remaining one, *feature selection*, is the main concern of my thesis. After the feature extraction steps, we have too many features even for a single frame in speech data. Despite the bulk of features seems good, irrelevant and redundant features influence the success of model negatively. To construct a good learner, we have to find an optimal feature set relevant to our task.

2.2. Related Methods

2.2.1. Canonical Correlation Analysis

Proposed by Hotelling [34], CCA seeks to maximize the mutual correlation between two sets of variables by finding linear projections for each set. Mathematically , CCA seeks to maximize the mutual correlation between two views of the same semantic phenomenon (e. g. audio and video of a speech) denoted $X \in \mathbb{R}^{n \times d}$ and $Y \in \mathbb{R}^{n \times p}$, where n denote the number of paired samples, via:

$$\rho(X, Y) = \max_{w,v} corr(w^T X, v^T Y), \tag{2.1}$$

where "corr" corresponds to Pearson's correlation, w and v correspond to the projection vectors of X and Y, respectively. Let C_{XY} denote the cross-set covariance between the sets X and Y, and similarly let C_{XX} denote within set covariance for X. The problem given in Equation 2.1 can be re-formulated as:

$$\rho(X, Y) = \sup_{w,v} \frac{w^T C_{XY} v}{\sqrt{w^T C_{XX} w \cdot v^T C_{YY} v}}. \tag{2.2}$$

The formulation in Equation 2.2 can be converted into a generalized eigenproblem for both projections (i. e. w and v), the solution can be shown [35] to have the form of:

$$C_{XX}^{-1} C_{XY} C_{YY}^{-1} C_{YX} w = \lambda w, \tag{2.3}$$

where the correlation appears to be the square root of eigenvalue:

$$\rho(X, Y) = \sqrt{\lambda}. \tag{2.4}$$

To attain maximal correlation, the eigenvector corresponding to the largest eigenvalue in Equation 2.3 should be selected. Similarly, by restricting the new vectors to be uncorrelated with the previous ones, it can be shown that the projection matrices for

each set are spanned by the k eigenvectors corresponding to the k largest eigenvalues. In short, when CCA is applied between X and Y we get:

$$[W, V, r, U_X, U_Y] = CCA(X, Y), \tag{2.5}$$

where W and V are composed of (sorted) eigenvectors from the eigenproblem in Equation 2.3, r is the m dimensional vector of canonical correlations given in Equation 2.4 while U_X and U_Y are the covariates. In other words, $U_X = X \times W$, when features in X are mean removed. The relationship between the canonical correlation and the corresponding covariates is given by the the Pearson's Correlation Coefficient (PCC):

$$\rho^i = PCC(U_X^i, U_Y^i), \tag{2.6}$$

where i indexes the column. It is important to note that the maximum number of covariates m in U_X and U_Y are limited with the matrix rank of X and Y:

$$m = min(rank(X), rank(Y)) \tag{2.7}$$

The non-linear version of CCA using the *kernel trick* is known as KCCA [35]. Also, Deep CCA (DCCA) is an efficient deep neural network alternative to KCCA [36].

2.2.2. Local Fisher Discriminant Analysis

It is known that when classes are multimodal, FDA faces anomalies [37]. It is important to preserve the local structure in the embedded space while trying to maximize the class separability. To retain the multimodality in the target space without regarding the classes, Locality Preserving Projection (LPP) [38] is introduced as an alternative to PCA. The approach uses the affinity matrix idea to weight (softly mask) the projections. This idea inspired Sugiyama to extend traditional FDA to Local FDA

by first reformulating the scatter matrices [39]:

$$S^w = 1/2 \sum_{i,j}^{n} A^w_{i,j}(x_i - x_j)(x_i - x_j)', \tag{2.8}$$

$$S^b = 1/2 \sum_{i,j}^{n} A^b_{i,j}(x_i - x_j)(x_i - x_j)', \tag{2.9}$$

where $(')$ denotes transpose and

$$A^w_{i,j} = \begin{cases} 1/n_c & \text{if } y_i = y_j = c, \\ 0 & \text{if } y_i \neq y_j, \end{cases} \tag{2.10}$$

$$A^b_{i,j} = \begin{cases} 1/n - 1/n_c & \text{if } y_i = y_j = c, \\ 1/n & \text{if } y_i \neq y_j, \end{cases} \tag{2.11}$$

Here the affinity matrices do not contain locality information but class information. To obtain LFDA we have [39]:

$$\overline{\mathcal{S}}^w = 1/2 \sum_{i,j}^{n} \overline{\mathcal{A}}^w_{i,j}(x_i - x_j)(x_i - x_j)', \tag{2.12}$$

$$\overline{\mathcal{S}}^b = 1/2 \sum_{i,j}^{n} \overline{\mathcal{A}}^b_{i,j}(x_i - x_j)(x_i - x_j)', \tag{2.13}$$

and localized discriminative affinity matrices are defined as

$$\overline{\mathcal{A}}^w_{i,j} = \begin{cases} A_{i,j}/n_c & \text{if } y_i = y_j = c, \\ 0 & \text{if } y_i \neq y_j, \end{cases} \tag{2.14}$$

$$\overline{\mathcal{A}}^b_{i,j} = \begin{cases} A_{i,j}(1/n - 1/n_c) & \text{if } y_i = y_j = c, \\ 1/n & \text{if } y_i \neq y_j, \end{cases} \tag{2.15}$$

where $A_{i,j}$ is the $n \times n$ regular affinity matrix keeping the unsupervised locality information. $A_{i,j}$ can simply be composed of 1s for k-nearest neighbors for each instance and 0s for the rest. It is also possible to adopt a localized measure where the distance to the k-th nearest neighbor is used as bandwidth in Gaussian similarity. Let D denote the $n \times n$ Euclidean distance matrix of samples, d_k is the n dimensional vector keeping

the square root of the Euclidean distance of each sample to its k-th neighbor, and $M./L$ denote the element-wise division, we can obtain a smoother affinity matrix A via:

$$L = d_k d_k', \tag{2.16}$$

$$A = \exp(-D./L). \tag{2.17}$$

Once the scatter matrices are computed, the regular FDA eigenproblem can be used to obtain the discriminative projection:

$$\overline{\mathcal{S}}^b W = \Lambda \overline{\mathcal{S}}^w W. \tag{2.18}$$

2.2.3. Extreme Learning Machines

Extreme Learning Machine (ELM) was first introduced a decade earlier [40] as a fast alternative training method for Single Layer Feedforward Networks (SLFNs). The rigorous theory of the ELM paradigm is presented in 2006 by Huang *et al.* [41], where the authors compare the performance of ELM, SVM, and Back Propagation (BP) learning based SLFN in terms of training time and accuracy. The basic ELM paradigm has matured over the years to provide a unified framework for regression and classification; and is related to generalized SLFN class including Least Square SVM (LSSVM) [42, 43]. Due to fast and accurate results obtained via ELMs, the method is applied in many real life tasks ranging from gesture recognition to representational learning [44, 45]. In this section, we provide a brief introduction to the paradigm.

The argument of basic ELM introduced by Huang *et al.* is that the first layer (input layer) weights and biases of a neural network classifier do not depend on data and can be randomly generated; the second layer (output weights) can be effectively and efficiently solved via least squares [41]. It can be thought that the input layer carried out unsupervised feature mapping, then the activation function outputs (the output matrix) is subjected to a supervised learning procedure. Let $\mathbf{x} \in \mathbb{R}^d$ denote

an input sample, $\mathbf{h}(\mathbf{x}) \in \mathbb{R}^p$ denote the hidden node output. Similarly, let $\mathbf{X} \in \mathbb{R}^{n \times d}$ denote the dataset and $\mathbf{H} \in \mathbb{R}^{n \times p}$ denote the hidden node output matrix. The hidden node activation via randomly generated mapping matrix $\mathbf{W}$ and bias vector $\mathbf{b}$ is defined as in regular SLFN:

$$H(l,t) = h_l(\mathbf{x^t}) = g(\mathbf{x^t}, \mathbf{w_l}, b_l), l = 1, ..., L, t = 1, ..., N, \tag{2.19}$$

where nonlinear activation function $g()$ can be any infinitely differentiable bounded function [41]. A common choice for $g()$ is sigmoid function:

$$g(\mathbf{x}, \mathbf{a}, b) = \frac{1}{1 + exp\left(-(\mathbf{a} \cdot \mathbf{x} + b)\right)}. \tag{2.20}$$

ELM proposes an unsupervised, even random generation of hidden node output matrix $\mathbf{H}$. The actual learning takes place in the second layer between $\mathbf{H}$ and the label matrix $\mathbf{T}$. $\mathbf{T}$ is composed of continuous annotations in case of regression therefore is a vector. In the case of M-class classification, $\mathbf{T}$ is represented in one vs. all coding

$$\mathbf{T}_{i,m} = \begin{cases} +1 & \text{if } y_i = m, \\ -1 & \text{if } y_i \neq m. \end{cases} \tag{2.21}$$

The second level weights β are learned by least squares solution to a set of linear equations $\mathbf{H}\beta = \mathbf{T}$. Proving first that random projections and nonlinear mapping with $L \leq N$ result in a full rank H, the output weights can be learned via

$$\beta = \mathbf{H}^{\ddagger}\mathbf{T}, \tag{2.22}$$

where $\mathbf{H}^{\ddagger}$ is the Moore-Penrose generalized inverse [46] that gives not only minimum L_2 norm solution to $||\mathbf{H}\beta - \mathbf{T}||$, but also minimizes the norm of projection $||\beta||$. The use of this special generalized inverse is motivated by Barlett's theory stating that for networks approximating an arbitrarily small training error, the smaller the norm of weights is, the better the generalization capability of the network [47]. The universal

approximation and classification capability of ELMs have been rigorously discussed in the literature (cf. [43]), and are beyond the scope of this paper. However, it is important to mention that ELM is related to Least Square SVMs via the following output weight learning formulation:

$$\beta = \mathbf{H}^T(\frac{\mathbf{I}}{C} + \mathbf{H}\mathbf{H}^T)^{-1}\mathbf{T}, \tag{2.23}$$

where $\mathbf{I}$ is $N \times N$ identity matrix, and C used to regularize the linear kernel $\mathbf{H}\mathbf{H}^T$ is indeed the complexity parameter of LSSVM [42]. The approach is extended to use any valid kernel. A popular choice for kernel function is Gaussian (RBF):

$$K(\mathbf{x_k}, \mathbf{x_l}) = \phi(\mathbf{x_k}) \cdot \phi(\mathbf{x_l}) = exp(-\frac{||\mathbf{x_k} - \mathbf{x_l}||}{\sigma^2}) \tag{2.24}$$

In both (basic and kernel) approaches, the prediction of $\mathbf{x}$ is given via $\hat{\mathbf{y}} = \mathbf{h}(\mathbf{x})\beta$. In case of multi-class classification the class with maximum score in $\hat{\mathbf{y}}$ is selected. In our study we utilize kernel version of ELM.

2.3. Literature Review

The feature selection methods aim to automatically select a useful subset of signal features to gain better classification performance than the large, non-selective baseline feature set. By effective feature selection, we will benefit from followings [48, 49]

- Enhanced classification performance due to the removal of noisy or unreliable features.
- Lower computational costs in the final system due to reduced dimensionality in feature extraction, model training and classification.
- Simpler classifiers with less input variables, which of-ten leads to better generalization ability towards new samples.
- Better hands-on understanding of the classification problem through discovery of

relevant and irrelevant features.

As we mentioned previously, many machine learning system suffers from 'curse of dimensionality problem' where data is to sparse with regard to high-dimensional feature space. This effect causes over fitting of trained model on data and decrease reliability of it. The solution is the selection of small but robust feature set by applying feature selection methods. In literature, feature selection methods gathered under three main groups: wrapper methods, filter methods and embedded methods.

Since the main goal is the find feature set which is optimal with underlying model for task at hand, we can select features based on a criterion function c, such as G(S, D, M) = c, where S denotes the feature set, D denotes the data and M is the underlying model applied in the task. This criterion function can be based on overall classification performance as in *wrapper methods* or some heuristic measures as in *filter methods*. Even this approach seems applicable, it has already two drawbacks. The first is the expense of search. The number of possible feature subsets grows exponentially as a function of the initial feature pool size so; search for the best feature set requires intelligent search algorithms or tricks. The second is the generalization problem. Since the criterion function use a limited size subset of complete data, there is no guarantee exist that the location of local maxima of criterion function with respect to S will remain same for the unused data.

In *wrapper methods*, the success of the selected features measured by the criterion function based on the overall classification/regression performance of the underlying model with provided data set. So, features selected or eliminated based on the value of criterion function. Sequential Backward Elimination (SBE) and Sequential Forward Selection (SFS) are the basis wrapper methods.

Sequential Backward Elimination (SBE), proposed by Marill and Green [50], starts with the complete feature set and sequentially eliminates features whose absence results in the best score.

Sequential Forward Selection (SFS), proposed by Whitney [51], works in the opposite direction as the names imply. It starts with an empty set and sequentially insert new features whose addition results in the best score.

The score of the feature insertion or deletion measured with respect to the previously selected feature set using a hill climbing scheme. Thus, these 'greedy' methods do not guarantee to examine all possible subset. The finding optimal feature is depending on the previously selected features. For both methods, newly selected feature set is a subset of a larger feature set seen earlier or later. This is called as '*nesting problem*' and researchers developed a new method, Sequential Forward Floating Search (SFFS), to overcome this issue [52]. This method is widely used for feature selection in related works. The main enhancement of SFFS is, it carries out sequential feature forward selection but also backward elimination. By that way, we can get rid of nesting problem and also have a greater chance to examine all possible combinations.

In *filter methods*, the role of classifier M in wrapper methods replaced with some heuristic measures in the criterion function. Common measures are mutual information, Pearson correlation, Mahalanobis distance. In simplest filter methods, features assigned scores individually based on the criterion function and then ranked. Feature subset either created by selecting distinct number of features from ranked features or applying some threshold. The other filtering methods are feature subset selection methods (Kohavi and John, 1997; Theodoridis and Koutroumbas, 2003). Correlation-based feature selection (CFS) [53] and the minimum Redundancy Maximum relevance (mRMR) approach [54] are well known examples. CFS and mRMR analyze correlation and mutual information, respectively. Both attempt to maximize dependence between the features and the class information, while simultaneously minimizing it between features in the selected feature set. CFS measures the heuristic merit between a feature set S and target t via [53]:

$$r_{S,t} = \frac{k\overline{r_{ti}}}{\sqrt{k + k(k-1)\overline{r_{ii}}}}, \tag{2.25}$$

where k is number of features, $\overline{r_{ti}}$ denote average correlation between the features in the subset and the target variable, and the term $\overline{r_{ii}}$ denote average inter-correlation between features. Hall (1999) proposes several measures of dependence to compute feature-feature and feature-target merits of a subset. When the target variable is continuous, Pearson's correlation coefficient is used. In the approach used in this thesis, Equation 2.25 is simplified, keeping the notion of high relevance low redundancy. CFS uses symmetrical uncertainty as the correlation measure when the class is nominal. A measure based on information theory estimates the degree of dependency between nominal features. Suppose X and Y are discrete random variables, Equations 2.26 and 2.27 give the entropy of Y before and after observing X.

$$H(Y) = -\sum_{y \in Y} p(y) \log_2 p(y), \tag{2.26}$$

$$H(Y|X) = -\sum_{x \in X} p(x) \sum_{y \in Y} p(y|x) \log_2 p(y|x). \tag{2.27}$$

The amount by which the entropy of Y decreases reflects additional information about Y provided by X and is called the information gain [55], or, alternatively, mutual information [56]. Mutual information is given in equation 2.28

$$\begin{aligned} MI &= H(Y) - H(Y|X) \\ &= H(X) - H(X|Y) \\ &= H(Y) + H(X) - H(X,Y). \end{aligned} \tag{2.28}$$

Information gain is a symmetrical measure that is, the amount of information gained about Y after observing X is equal to the amount of information gained about X after observing Y . Symmetry is a desirable property for a measure of feature-feature intercorrelation to have. Unfortunately, information gain is biased in favor of features with more values. Furthermore, the correlations in Equation 2.25 should

be normalized to ensure they are comparable and have the same affect. Symmetrical uncertainty [57] compensates for information gain's bias toward attributes with more values and normalizes its value to the range [0, 1]:

$$symetrical\ uncertainity = 2.0 \times \left[\frac{MI}{H(X) + H(Y)}\right] \tag{2.29}$$

Similarly, mRMR drives the feature selection in a set X, at step k maximizing the difference or ratio between relevance and redundancy terms [54]:

$$\max_{x_j \in X - S_{k-1}} \left[MI(x_j, t) - \frac{1}{k-1} \sum_{x_i \in S_{k-1}} MI(x_j, x_i) \right], \tag{2.30}$$

where $MI(x, y)$ is mutual information between random variables x and y. In KCCAmRMR, Sakar *et al.* [58] improved mRMR feature selection using correlated functions of variables (i. e. projections attained by CCA) weighted with corresponding correlations with the target variable. In this work, MI is completely replaced with CCA.

Nowadays, the computational paralinguistic is very hot topic and many researchers aim to develop a novel system in this area. But recently, the focus on system design replaced with search for an efficient feature selection method. In INTERSPEECH 2013 Challenge,despite the brute-forced baseline feature sets provide the-state-of-the-art results, in some recent work the use of a small set of domain-knowledge inspired features is shown to outperform the baseline set [59]. Important to note is that although extracted via an alternative way, the knowledge-inspired set used was a small subset of the baseline features. This finding motivates the need for a robust feature subset. In Table 2.1, a non-exhaustive summary of recent literature work on feature selection in computational paralinguistics is given.

Table 2.1. A Summary of Feature Selection (FS) Methods in Computational Paralinguistics.

Work	**Paralinguistic Task**	**Method**
Torres *et al.*(2006) [60]	Depression	GP-Based Two-Stage FS
Park *et al.*(2006) [61]	Emotion	Interactive FS
Torres *et al.*(2007) [62]	Depression	GA-Based FS
Espinosa *et al.*(2011) [63]	Emotion	Bilingual Acoustic FS
Giannoulis and Potamianos (2012) [64]	Emotion	mRMR + SBE
Räsänen *et al.* (2013) [65]	Autism, Emotion and Level of conflict	Random Subset FS
Kirchhoff *et al.* (2013) [66]	Autism	Submodular FS
Moore *et al.*(2014) [67]	Emotion	Correlation FS
Kaya *et al.*(2014) [32]	Depression	CCA based FS
Bejani *et al.*(2014) [68]	Emotion	ANOVA based FS
Kim *et al.*(2014) [69]	Level of Conflict	Automatic Relevance Determination

Although the authors apply different feature selection methods from each other, the main purposes are same for all: avoiding curse of dimensionality problem , eliminating redundant features and finding relevant features. In [60], authors apply genetic programming approach for feature selection. The authors state that the wrapper methods suffer from the computational expense. Hence, instead of using classifier for each trial, they use it as a fitness function. While they apply genetic programming to select optimal features, they use classifier in the fitness function. In this way, they do feature selection and classifier design simultaneously. The evolutionary method they apply is called as GPFS. Feature selection in the GPFS begins during the generation of the initial population, where a feature set is randomly generated for each individual.Once the initial population is created, the fitness of each individual is computed. The fitness shows the classification accuracy. And then, they select the two individual which have best fitness score and exposed them to crossover operations. The population evolve with the newly created feature sets until convergence.In [62], authors extend their previous GA-based work [60]. The authors stated that the previous approach suffer from considering and evaluating features independently from each other. In this new approach, they apply correlation based clustering and Two-stage Genetic Algorithm, respectively. The main goal in their work is to find an optimal clustering which shows best classification accuracy. A clustering is done based on the distance measure between features. Clustering can be seen as filtering and upon completion of this step, they execute two stage-Genetic algorithm. In the first stage, they try to select clusters which will be taken into classification phase. In the second stage, they analyze selected clusters and search for a feature which best represents the cluster.

In [63], the authors search for acoustic features that are good enough to estimate the emotional state based on the voice of a person independently from which language they use.They use Linear Floating Forward Selection (LFFS) [70] to select features. This method makes a hill-climbing search. It starts with the initial feature set, evaluates all possible inclusions of a single attribute and select the one which have best evolutionary result. The search ends where there are no attribute to improve evaluation.In their experiments, they use LFFS with Fixed Width mode. In this mode, initially k features are selected and the others are remains. At each step, features

selected to be added replaced with the one feature from remaining set. In [61], authors developed a new method, 'Interactive Feature Selection'(IFS) for feature selection and the results of IFS were applied to their own emotional recognition system. The IFS based on reinforcement learning and requires responses from human users. This algorithm is inductive and based on the rationale of correlation. This is finding a good feature subset which contains features highly correlated with a class but uncorrelated which each other. Their method starts with the full feature set. Every time a new feature set and emotion identifier is inputted from user, it assigns a return sign (+1 if an old emotion identifier equals the new emotion identifier, - otherwise). The product of return sign and the difference of each other is stored. This iteration is repeated during one episode. After the episode, each subset feature set is applied to objective function and evaluation result is stored. If the next evaluation result is worse than previous, the worst feature of selected feature set will be replaced with the best feature among those that were not selected.In [66], the authors apply a novel feature selection technique.Their method based on submodular function optimization. They tested their method on INTERSPEECH 2013 Autism-Challenge and achieve significant success relative to baseline. In [64], the authors use Two-stage feature selection method. They want to take advantages of wrapper and filter method together. Hence, they use both types of feature selection methods respectively. In the first stage they apply mRMR filter method. In the second stage, they apply backward feature selection wrapper strategy. In [67], the authors apply Correlation Feature Selection technique. In [68], the authors claim that they use ANOVA (Analysis of Variance) technique to select a feature subset. However, they do not give detailed information about how they use ANOVA for feature selection in their work. In [69], the authors utilize a Bayesian approach for regression of level of conflict, where feature selection is done by ARD.

In [65], a new feature selection method, Random Subset Feature Selection is introduced. At each time, they select a random feature set by applying uniform distribution on full feature set. Then, they measure relevance of each feature based on the performance of the subset that the feature participates in. To compute the relevance, the authors increase the points of features participating in a set providing higher than average performance by a predefined value p, and similarly reduce the same amount

for the features performing lower than the average. In this wrapper method, the performance is measures with k-nearest neighbors classifier. After 300 000 iterations, the features are ranked with respect to their points. Despite its good performance on challenging tasks, this method is weak in three aspects (i) Efficiency: It requires hundreds of thousands classification iterations and hence does not scale to large data (ii) Feature weighting: The relative weights of features in a classification are not taken into account, all get the same reward/punishment. (iii) Feature group weighting: The method rewards/punishes based on relative performance to average but how better/worse is not considered. In this thesis, the method proposed weights both the group and the participating features separately in a principled way using CCA.

3. PROPOSED METHOD

3.1. Discriminative Projection Based Filters

The proposed method in this thesis extends the recent work of Kaya *et al.* [32] by applying the discriminative projection based ranking to random subsets of a large feature set. The main idea behind the CCA based filter in [32] is as follows. When all features on one view are subjected to CCA against the labels on the other view, the absolute value of the projection matrix W can be used to rank the features. The application to regression is straightforward since the resulting matrix is $n \times 1$, therefore a vector. It can be applied in the same way to Two-class classification where the classes can be denoted with 0 and 1 in the target vector. For $C > 2$, we can use the canonical correlation value (ρ^i) to weight the corresponding projection column (eigenvector W^i). In short, SLCCA-Filter algorithm, which inputs the dataset $X \in \mathbb{R}^{n \times d}$ and label matrix $T \in \{0,1\}^{n \times (C)}$; and outputs feature ranking R is given as:

$$
\begin{aligned}
[W, V, \rho, U_X, U_Y] &= CCA(X, T), && (3.1)\\
H &= \sum_{i=1}^{m} abs(W^i)\rho^i, && (3.2)\\
R &= \text{sort(H,'descend')}, && (3.3)
\end{aligned}
$$

where as noted earlier $m = min(rank(X), rank(Y))$, and the 1-of-C coded label matrix T is defined as

$$
T_{i,c} = \begin{cases} 1 & \text{if } y_i = c, \\ 0 & \text{if } y_i \neq c. \end{cases} \tag{3.4}
$$

Since C classes have $C-1$ degrees of freedom, the rank of matrix T is $C-1$. Therefore it is possible to remove any of the columns from 1-of-C coded matrix. The filter can be applied to FDA or LFDA in a similar manner, where instead of the canonical correlation value square root of the corresponding eigenvalue λ^i is used.

3.2. Random Discriminative Projection Based Filters

As mentioned above the proposed method extends the idea of discriminative projection based filters explained in Section 3.1. Though it looks efficient and valuable to suppress redundant features, SLCCA-Filter has an important drawback, which gives the motivation to this thesis study: the number of non-zero weight features in the projection is upper-bounded by the rank of the data matrix. The reason for this lies in the pseudo-inversion of the singular covariance matrix. Therefore, when dimensionality is very high, many features that have unique information about the target variable will be omitted. This problem is referred as *irrelevant redundancy* in the literature [58].

By means of random sampling of features, it is possible to evaluate feature relevance/redundancy in different conditions and aggregate them to obtain a final ranking. While the absolute value of feature projection matrix (eigenvectors of the corresponding eigenproblem) provide information about feature level saliency (driven to zero if the feature is redundant or irrelevant), the square root of the eigenvalue in a discriminative projection is used to weight how good the feature group collectively performs. As mention earlier, in case of $K > 2$ class classification the result is a $K - 1$ dimensional projection. Here feature saliency in the projection matrix is weighted by the corresponding eigenvalues (canonical correlation statistic ρ in CCA) also providing the saliency for randomly chosen feature groups to be aggregated.

A Toy Example. Before proceeding to the details of the algorithm, the redundancy elimination and relevance maximization properties of this method will be discussed with a toy example. Suppose we have three randomly generated vectors a, b and c drawn independently from standard normal distribution. All vectors have 1000×1 dimensions. Let's define a redundant feature d, the target variable t and the dataset X as follows: $d = a/2$, $t = 2 * a + b$, and $X = [a\ b\ c\ d]$, respectively. The d vector shows there exists a linear dependency between a and d. This means that one of them is redundant. In addition, formulation of the target variable shows that c is irrelevant. In the light of our motivation, the projection matrix of canonical correlation between

X and t represents the feature ranking among a, b, c and d variables. Due to the linear dependence among a and d, one of them is redundant therefore its weight should be set to 0. After when X is subjected to CCA against t, we get the feature projection matrix $A = [-0.8988, -0.4494, 8.6264 \times 10^{-17}, 0]$ where the weight of d is equal to 0. The highest absolute weight is with a that is the biggest constituent of target variable. We also see that the irrelevant variable c has almost zero weight. This example shows that the method provides redundancy elimination and relevance maximization, concurrently. However, linear dependence of features (matrix rank issue) may hinder obtaining a more relevant and less redundant compact set. This is weakness is the main motivation of this thesis study.

In Figure 3.2, psedo-code of the proposed method, Random SLCCA initial version is given. Basically, we randomly generate feature indices in the size of total feature dimension. Then, we select a predefined number of features to project. We locate the selected features in original data and store as current training data matrix. After that, we calculate the canonical correlation between data and target labels. The projection matrix and canonical correlation values are used to form weighted feature saliency vector. We perform these steps at each iteration and sum the weighted absolute projections. At the end, we sort the total weighted projection vector to get ranking of features.

In the proposed method, maximum dimensions to select randomly at each iteration is a critical parameter. The results of feature selection with small D were not found to be good. We observed that this method requires a high number of iterations T and randomly drawn features $maxD$ to perform as good as baseline SLCCA. This is undesired because we wish to keep the method scalable while increasing the accuracy. In the first case, we select D, the dimension of randomly selected features to project, is small as possible. The problem in this method is that we can not *touch* all of the features and there exist a high possibility of skipping relevant features with small values of $maxD$ and K. In Figure 3.2 a simulated plot of the probability of skipping features versus the number of iterations is shown for a set of $maxD$ in a imaginary dataset of 10 000 features. We see, for example, that if we randomly draw 100 features to apply

Input:

X: NxD dimensional data matrix

T: target matrix N x (C-1) in classification , N x 1 in regression

K : Number of iterations

maxD : maximum dimensions to select randomly at each iteration

Require X, T , maxD and K as input.

for $i = 1$ to **K do**

  RandFeats $\Leftarrow$ randperm(D);

  FeatsIdxs $\Leftarrow$ RandFeats(maxD);

  $\mathbf{X_{rand}} \Leftarrow$ X(:, FeatsIdxs) ;

  Apply CCA on selected feature set $\mathbf{X_{rand}}$ and **target** using Equation 3.1 ;

  Compute the weight vector $\mathbf{H_i}$ using Equation 3.2 ;

  Zero pad $\mathbf{H_i}$ for unselected features to obtain D dimensional saliency vector $\mathbf{w_i}$;

  Cumulate weighted features in vector $\mathbf{W}$: $\mathbf{W} \Leftarrow \mathbf{W} + \mathbf{w_i}$;

end for

Obtain the sorted saliency,$\mathbf{W_s}$ and feature ranking **R** by applying Equation 3.3 on **W** ;

Figure 3.1. Random SLCCA Algorithm Initial Version.

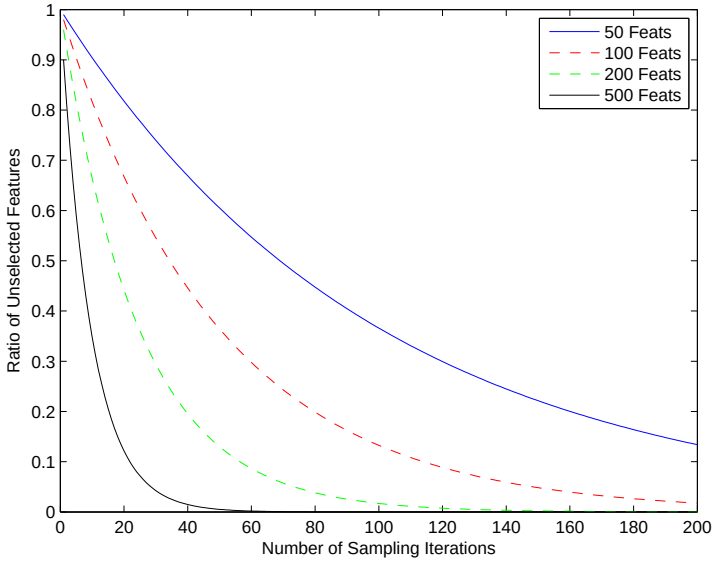

Figure 3.2. Probability of Skipping Features in Sampling with Replacement vs Number of Iterations in an imaginary dataset having 10 000 features.

SLCCA, at 40 iterations still half of the features are untouched. This led to the second version where we project both the randomly selected set and its complement set.

In the second version, we discriminatingly project the randomly selected feature set and its complement but, the results were not good enough again. In this case, the difference in dimensions between feature sets to be projected caused problems. If their dimensions are not close to each other, the weights assigned to features in projection matrix vary highly. Suppose we have data matrix X with dimensions $N \times 6373$, where N is the number of samples and 6373[2] is the feature dimension. If we select $maxD$ is equal to 100, we got N x 100 as randomly selected feature set while N x 6273 for the remaining set. After the projection of both feature sets, we can see that the average weights for first 10 variables are around to 10^5 in randomly selected feature set, as opposed to 10^{-1} in the complement set. This difference arose from the gap in dimensions and can cause incompatibility.

As an alternative for previous two, we select $maxD$ as half of the original feature dimensionality in third version. With this approach, we can both access all features at

[2]This is the INTERSPEECH 2013 baseline feature set dimensionality.

each iteration during projection and also achieve to get compatible feature weights in projection matrix. This approach effectively eliminates the $maxD$ parameter. In the remaining part of this thesis, I report results with $maxD = D/2$. The final version of the algorithm is given in Figure 3.3.

Input:

X: NxD dimensional data matrix

T: target matrix N x (C-1) in classification , N x 1 in regression

K : Number of iterations

maxD : maximum dimensions to select randomly at each iteration

Require X, T and K as input.

maxD $\Leftarrow$ D / 2;

for $i = 1$ to **K do**

  RandFeats $\Leftarrow$ randperm(D);

  FeatsIdxs $\Leftarrow$ RandFeats(maxD);

  $\mathbf{X_{rand}} \Leftarrow$ X(:, FeatsIdxs);

  $\overline{\mathbf{X_{rand}}} \Leftarrow$ X(:, $\overline{FeatsIdxs}$);

  Apply CCA on selected feature set **X-rand** and **target** using Equation 3.1;

  Apply CCA on complement set $\overline{\mathbf{X_{rand}}}$ and **target** using Equation 3.1;

  Compute the weight vectors $\mathbf{H_i}$ and $\overline{\mathbf{H_i}}$ for each projection using Equation 3.2;

  Combine $\mathbf{H_i}$ and $\overline{\mathbf{H_i}}$ to obtain D dimensional saliency vector $\mathbf{w_i}$;

  Cumulate weighted features in vector $\mathbf{W}$: $\mathbf{W} \Leftarrow \mathbf{W} + \mathbf{w_i}$;

end for

Obtain the sorted saliency,$\mathbf{W_s}$ and feature ranking **R** by applying Equation 3.3 on **W** ;

Figure 3.3. Random SLCCA Algorithm Final Version.

4. EXPERIMENTS AND RESULTS

4.1. AVEC 2013 Depression Corpus

According to depression literature, people who suffer from mood disorders can be recognized by their behaviors in social interaction. The psychologists and psychiatrists consider the vocal and facial clues during diagnosis. For instance, depression could result in expressive behavior such as dampened facial expressions, avoiding eye contact, and using short sentences with flat intonation. Hence the mental health problem often affects people at working age; this illness causes significant recession in economy, justice and education system. This makes diagnosis and cure of the mental illness important for our environment. Although the recent improvements in both health and technology areas, automatic measurement and assessment of mood disorders have not been deployed as a real word application yet. AVEC 2013 Depression Challenge and Corpus [14] addresses this situation with an aim to develop predictive systems for use of mental health practitioners.

AVEC 2013 uses a subset of the audio-visual depressive language corpus (AVDLC), which includes 340 video clips of subjects performing a Human-Computer Interaction task while being recorded by a webcam and a microphone. In AVDLC, the total number of subjects is 292 and only one person appears per clip, i. e. some subjects feature in more than one clip. The speakers were recorded between one and four times, with a period of two weeks between the measurements. Table 4.1 summarizes basic statistics of the corpus [14]. Recorded behavior includes speaking out loud while solving a task, counting from 1 to 10, read speech (excerpts of a novel and a fable), singing in German, telling a story from the subjects' own past (the best event and a sad event from childhood). The depression levels were labeled per clip using Beck Depression Inventory-II (BDI-II) [71], a subjective self-reported 21 item multiple-choice inventory.

For the AVEC 2013 challenge, the recordings were split into three partitions: training, development, and test sets of 50 recordings each, respectively.

Table 4.1. Statistics of the AVDLC [14].

Property	**Statistic**
# of Clips	340
# of Subjects	292
Range of Clip Length	20-50 min.
Mean Clip Length	25 min.
Total Duration	240 hours
Age Range of Subjects	18-63 years
Mean±Std of Age of Subjects	31.5±12.3 years
BDI-II Score Range	0-45

4.1.1. AVEC 2013 Baseline Acoustic Feature Set

The AVEC 2013 audio baseline feature set consists of 2 268 features extracted using TUM's open source feature extractor openSMILE [31]. The features are composed of 32 energy and spectral related low-level descriptors (LLD) x 42 functionals, 6 voicing related LLD x 32 functionals, 32 delta coefficients of the energy/spectral LLD x 19 functionals, 6 delta coefficients of the voicing related LLD x 19 functionals, and 10 voiced/unvoiced durational features [14]. The complete list of LLDs and functionals are given in Tables 4.3 and 4.2, respectively.

The audio features are computed on short episodes of audio data. Since the Challenge dataset contains long continuous recordings, three segmentations have been performed: (i) voice activity detection (VAD) based (ii) overlapping short fixed length segments (3 seconds) and, (iii) overlapping long fixed length segments (20 seconds).

In short and long segmentation, the windows are shifted forward at a rate of one second. Functionals are then computed over each segment. Together with the per instance computation of functionals, the baseline feature set is provided in 4 versions to grasp relatively short-long acoustic characteristics of speech intended for depression and affect tasks. Both the challenge paper [14] and the benchmarking work of Kaya *et*

Table 4.2. Set of all 42 functionals. [1]Not applied to delta coefficient contours. [2]For delta coefficients the mean of only positive values is applied, otherwise the arithmetic mean is applied. [3]Not applied to voicing related LLD.

Statistical functionals (23)
(positive[2]) arithmetic mean, root quadratic mean,
standard deviation, flatness, skewness, kurtosis,
quartiles, inter-quartile ranges,
1 %, 99 % percentile, percentile range 1 %–99 %,
percentage of frames contour is above: minimum + 25%, 50%, and 90 % of the range,
percentage of frames contour is rising,
maximum, mean, minimum segment length[1,3],
standard deviation of segment length[1,3]
Regression functionals[1] (4)
linear regression slope,
corresponding approximation error (linear),
quadratic regression coefficient a,
approximation error (linear)
Local minima/maxima related functionals[1] (9)
mean and standard deviation of rising and falling slopes (minimum to maximum),
mean and standard deviation of inter maxima distances,
amplitude mean of maxima,
amplitude range of minima,
amplitude range of maxima
Other[1,3] (6)
LP gain, LPC 1 – 5

Table 4.3. AVEC 2013 low-level descriptors as given in [14].

Energy & spectral (32)
loudness (auditory model based)
zero crossing rate
energy in bands from 250 – 650 Hz, 1 kHz – 4 kHz
25 %, 50 %, 75 %, and 90 % spectral roll-off points,
spectral flux, entropy, variance, skewness, kurtosis,
psychoacousitc sharpness, harmonicity, flatness
MFCC 1-16
Voicing related (6)
F_0 (sub-harmonic summation, followed by Viterbi smoothing), probability of voicing
jitter, shimmer (local), jitter (delta: "jitter of jitter")
logarithmic Harmonics-to-Noise Ratio (logHNR)

al. [32] that use the baseline set report the best results with long segmented features. Therefore, in this thesis we focus on the long segmented baseline feature set. In long segmented set, there are 23439, 23087 and 23399 instances for training, development and test sets, respectively.

4.1.2. Experimental Results

In the experiments, the WEKA [72] implementation of CFS with "Best First" search as and SLCCA-Filter methods were used as independent benchmarks. Bagging-Tree (TreeBager) implementation in MATLAB is utilized as regressor on this corpus. The hyper-parameters of CFS was left as default, while a set of hyper-parameters was tested for SLCCA-Filter and Tree-Bagger. As detailed before, we followed the training, development and testing protocol of the challenge.

The performance measures used in this corpus are MAE and RMSE, where the latter is competition measure. The measures are defined as follows:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|\hat{y}_i - y_i|, \tag{4.1}$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2}, \tag{4.2}$$

where $\hat{y}_i$ and y_i denote the predicted and groundtruth scores, respectively.

In Table 4.4, we can see the comparison between SLCCA-Filter, CFS and AVEC 2013 Challenge baseline results. Kaya *et al.* report the state-of-the-art results using SLCCA-Filter on this corpus/sub-challenge [32]. The Table 4.5 shows the comparison between SLCCA-Filter and our method, SLCCA-Rand. Given results are obtained using 10-Tree bagger as a regressor. We observe that the best results are obtained using only 100 features as opposed to original dimensionality of 2 268. Moreover, the best result obtained here outperforms the one reported with benchmark method [32]. The experiment results for different tree number is plotted in Figure 4.1. We also compare the results with different classifiers by focusing on the most successful feature set range in Table 4.6.

Table 4.4. Development Set Performances of Benchmark Methods per Segmentation.

	Per Clip		**Long Segment**	
Method	**MAE**	**RMSE**	**MAE**	**RMSE**
All	9.75	11.89	7.93	10.24
CFS	9.30	11.46	8.24	10.22
SLCCA-Filter	8.92	11.00	7.84	10.22

In addition to these experiments, we investigate the effects of PCA on performance to ensure that it can reduce the computational complexity. We apply PCA on training data before the canonical correlation analysis in both methods SLCCA Filter and SLCCA Rand. In this case the redundancy elimination-relevance maximization process is divided into two. The unsupervised redundancy elimination is mostly done by PCA, then the relevance maximization is handled by CCA. To combine both for

Table 4.5. Experiment results with using 10 Tree Bagger and D=1100.

Method	**SLCCA-RAND**		**SLCCA-Filter**	
# of features	**MAE**	**RMSE**	**MAE**	**RMSE**
50	7.94	10.21	8.52	10.61
100	7.57	9.71	8.13	10.31
150	7.84	9.81	8.06	10.23
200	7.82	10.07	7.96	10.45
250	8.11	10.49	8.05	10.39
300	8.01	10.42	7.98	10.31
350	8.10	10.59	7.78	10.20
400	8.14	10.59	8.18	10.68

Table 4.6. Comparison of Best RMSE Performances of Bagging Tree(T=10), Elm Linear and Elm RBF Kernel (K=10 / 100).

Method	**Bagging Tree**	**ELM Linear**	**ELM RBF**	ELM RBF
# feats/Params	**(T=10)**	**-**	**(K=10)**	**(K=100)**
50	10.21	10.72	9.89	9.84
100	9.71	10.65	9.64	9.95
150	9.81	10.92	9.59	9.88
200	10.07	10.84	9.91	10.07

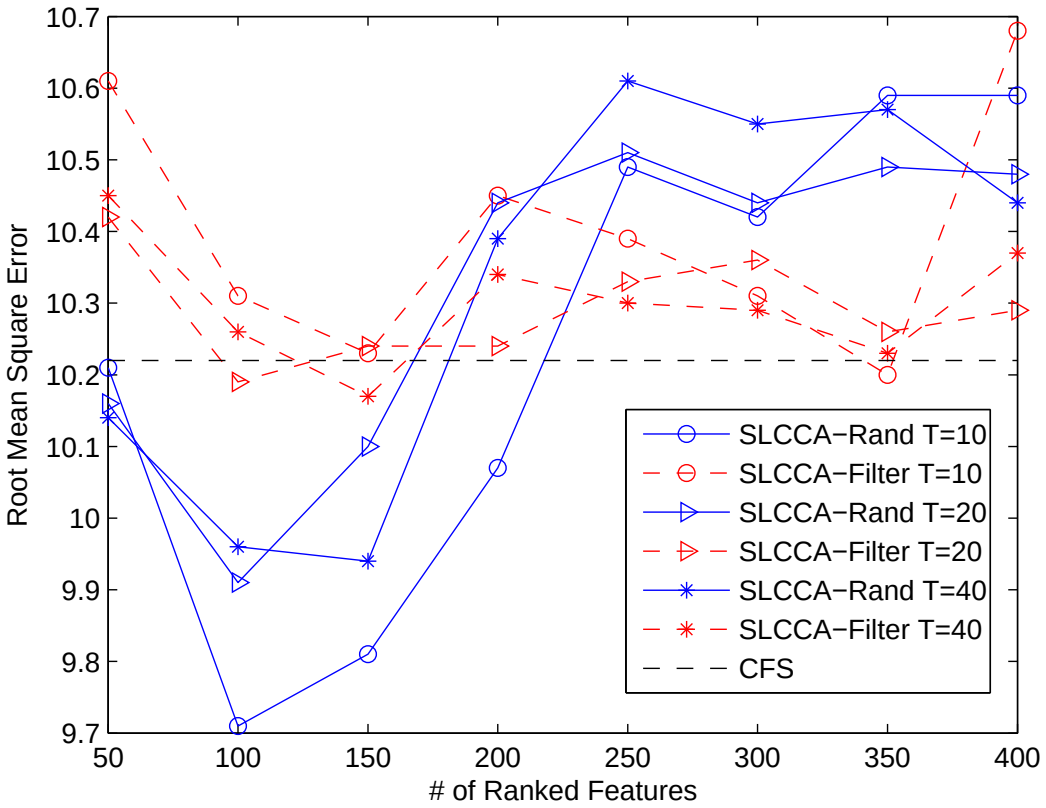

Figure 4.1. Regression Results Comparison Based on Tree Number.

selection of features in the original space we have an auxiliary matrix

$$\widehat{W} = W_{PCA} \times \Lambda_{PCA} \times W_{CCA} \times \rho_{CCA}, \tag{4.3}$$

where $W_{PCA/CCA}$ are projection matrices learned sequentially; Λ_{PCA} and ρ_{CCA} are diagonal matrices with corresponding eigenvalues/canonical correlations sorted in descending order. Note that we do not need to keep all eigenvectors for PCA. In this strategy, the feature ranking is obtained by sorting with respect to absolute value of $\widehat{W}$. The advantages of PCA are denoising/uncorrelating the data and reducing the computational load of CCA.

The table 4.7 shows the comparison of RMSE values for SLCCA Filter and PCASLCCA Filter methods by using Elm Linear. Experiment results for SLCCA Rand and PCASLCCA Rand methods are given in tables 4.8. The listed results indicate that using PCA before applying canonical correlation analysis increases the model succes for this range of selected feature number.

Table 4.7. Comparison of SLCCA Filter and PCASLCCA Filter with ELM Linear Kernel.

# of features	**SLCCA Filter**	**PCASLCCA Filter**
50	11.79	11.42
75	11.68	10.77
100	11.6	10.7
125	11.68	10.71
150	11.71	10.57
175	11.69	11.03
200	11.63	11.0
225	11.59	10.6
250	11.54	10.84
275	11.5	10.55
300	11.55	10.66
325	11.5	10.55
350	11.39	10.53
375	11.37	10.59
400	11.34	10.49

Table 4.8. Comparison of SLCCA Rand and PCASLCCA Rand by using ELM Linear Kernel.

# of features	**SLCCA Rand**	**PCASLCCA Rand**
50	11.79	11.29
75	11.68	10.7
100	11.60	10.68
125	11.68	10.80
150	11.71	10.51
175	11.69	10.67
200	11.63	10.82
225	11.59	10.67
250	11.54	10.72
275	11.50	10.35
300	11.55	10.32
325	11.50	10.59
350	11.39	10.55
375	11.37	10.59
400	11.34	10.52

4.2. INTERSPEECH 2013 Conflict Corpus

In response to the increased number of partitions to former Interspeech Challenges, they extend their scope by adding Conflict Corpus. The Conflict Sub-Challenge allows automatically analyzing group discussions with the aim of recognizing conflict. The works on this corpus are important since they improve paralinguistic researches with involving dyadic speech and speaker group analysis in realistic everyday communication.

The Interspeech 2013 Conflict Sub-Challenge [13] uses the "SSPNet Conflict Corpus"(SC^2) [73]. It contains 1 430 clips of 30 seconds extracted from 45 political debates televised in Switzerland. The clips are in French. The corpus includes 138 subjects in total: 23 females (1 moderator and 22 participants) and 133 males (3 moderators and 120 participants).The statistics about the corpus summarized in Table 4.9 .

Table 4.9. Statistics of the Conflict Corpus.

Property	**Statistic**
# of Clips	1430
# of Subjects	138
# of Female	23 (1 moderator , 22 participants)
#of Male	133 (3 moderator, 120 participants)
# of Political Debate	45
Total Duration	30 second
Conflict Score Range	(−10,+10)

The clips have been annotated following the process illustrated in [74] with respect to conflict level by roughly 550 assessors recruited via Amazon Mechanical Turk. Each clip is assigned a continuous conflict score in the range [10, +10], giving rise to a straightforward regression task. A binary classification task is created based on these labels, namely to classify into 'high' (> 0) or 'low' (< 0) level of conflict. The distribution among partitions is illustrated in Table 4.10 for classification.

Table 4.10. Partitioning of the SSPNet Conflict Corpus into train,development, and test sets for binary classification [13].

#	train	dev	test	total
low	471	127	226	824
high	322	113	171	606
total	793	240	397	1430

4.2.1. INTERSPEECH 2013 Baseline Acoustic Feature Set

The Interspeech 2013 Challenge baseline acoustic feature set was created by modifying the acoustic features set of previous challenge, Interspeech 2012 Spekar Trait Challenge [75]. The acoustic feature set from previous challenge created by using TUM's open-source openSMILE feature extractor and provide extracted feature sets on a per-chunk level and a configuration file to allow for additional frame-level feature extraction.The feature set consists of 4 energy related LLD, 54 spectral LLD and 6 voicing LLD.The complete list of functionals and LLDs are given in Table 4.12 and 4.11, respectively. Totaly, the previous challenge includes 6 125 features. In our challenge, they modified this feature set by improving voice quality features (jitter and shimmer), adding Viterbi smoothing for F0 and simplifying some applied functionals. Altogether, the 2013 COMPARE feature set contains 6 373 features.

Table 4.11. 65 provided low-level descriptors as given in [75].

4 energy related LLD
Sum of auditory spectrum (loudness)
Sum of RASTA-style filtered auditory spectrum
RMS Energy
Zero-Crossing Rate
54 Spectral LLD
RASTA-style auditory spectrum, bands 1-26 (0–8 kHz)
MFCC 1–14
Spectral energy 250–650 Hz, 1 k–4 kHz
Spectral Roll Off Point 0.25, 0.50, 0.75, 0.90
Spectral Flux, Entropy, Variance, Skewness, Kurtosis, Slope, Psychoacoustic Sharpness, Harmonicity
6 voicing related LLD F_0 by SHS + Viterbi smoothing, Probability of voicing logarithmic HNR, Jitter (local, delta), Shimmer (local)

Table 4.12. Applied functionals. [1] : arithmetic mean of LLD / positive Δ LLD [2]: only applied to voice related LLD. [3]: not applied to voice related LLD except F_0.[4]: only applied to F_0.

Functionals applied to LLD / Δ LLD
quartiles 1–3, 3 inter-quartile ranges
1 % percentile (≈min), 99 % percentile (≈ max) position of min / max
percentile range 1 %–99%
arithmetic mean[1], root quadratic mean
contour centroid, flatness
standard deviation, skewness, kurtosis
rel. duration LLD is above / below 25 / 50 / 75 / 90 % range
rel. duration LLD is rising / falling
rel. duration LLD has positive / negative curvature[2]
gain of linear prediction (LP), LP Coefficients 1–5
mean, max, min, std. dev. of segment length[3]
Functionals applied to LLD only
mean of peak distances
standard deviation of peak distances
mean value of peaks
mean value of peaks – arithmetic mean
mean / std.dev. of rising / falling slopes
mean / std.dev. of inter maxima distances
amplitude mean of maxima / minima
amplitude range of maxima
linear regression slope, offset, quadratic error
quadratic regression a, b, offset, quadratic error
percentage of non-zero frames [4]

4.2.2. Experimental Results

As in the Depression challenge, we used the WEKA [72] implementation of CFS with "Best First" search as and SLCCA-Filter methods as independent benchmarks for Conflict challenge. We used Unweighted Average Recall (UAR), which is the mean of individual recalls, as primary evaluation measure:

$$UAR = \frac{1}{C}\sum_{c=1}^{C} TP(c)/P(c), \tag{4.4}$$

where C is the number of classes; $TP(c)$ and $P(c)$ denote the number of true positive instances and total positive instances for class c, respectively.

For classification we applied Linear Support Vector Machine implementation in WEKA [72] and for each task, we choose the SVM complexity parameter C $\in \{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1, 10, 100\}$. We do classification on selected features ranked using both regression labels and classification labels.

The Figures 4.2 and 4.3 correspond to comparative performances of class-labels based vs. regression labels based raking learning using SLCCA-Rand and SLCCA-Filter methods, respectively. Noth figures show that the classification with features ranked by regression labels gives better UAR values than those ranked by class labels. The results in these figures are obtained by using Linear SVM as classifier. In the light of this comparison, we observe the advantage of regression labels for classification.

As mentioned before, we compared the SLCCA-Rand method with SLCCA-Filter and CFS. The baseline and CFS results are similar to each other and they have maximum UAR score of 79.1% and 74.5%, respectively. In Figure 4.4, we can see that the SLLCA-Rand method gives better UAR values than both SLCCA-Filter and baseline on Conflict data. Moreover, we see that the results obtained by the proposed method (SLCCA-Rand) yields a smoother (less fluctuating) trajectory with respect to number of ranked features. Thus, it is easier to estimate the optimal number of features.

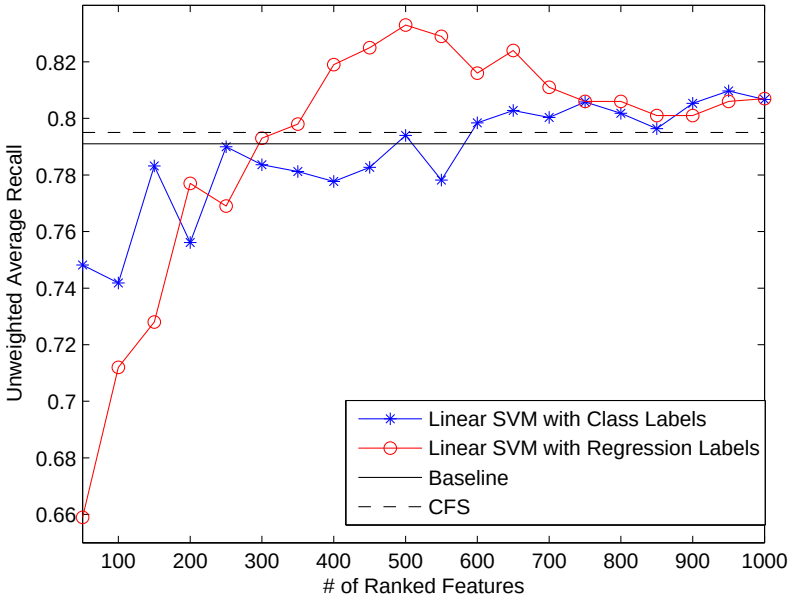

Figure 4.2. Comparison of Feature Ranking Learned from Regression Labels and Classification Labels - (SLCCA-Rand).

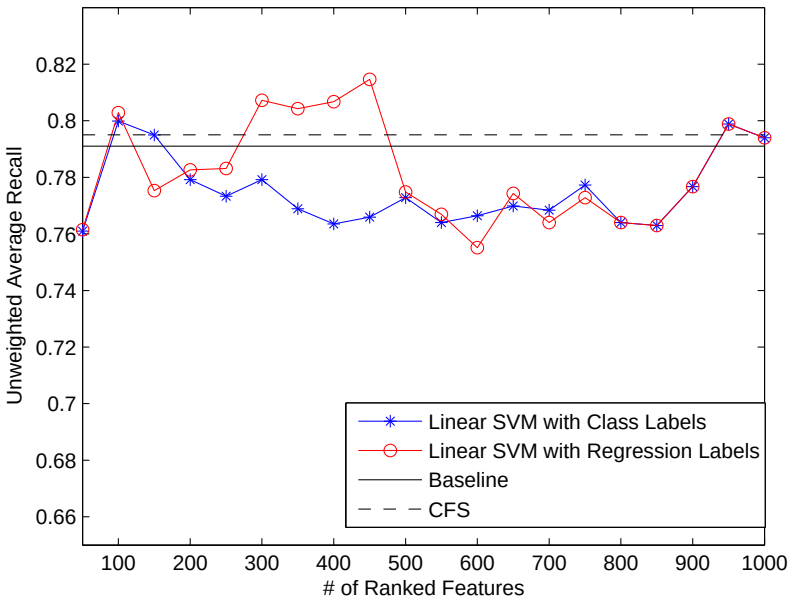

Figure 4.3. Comparison of Feature Ranking Learned from Regression Labels and Classification Labels - (SLCCA-Filter).

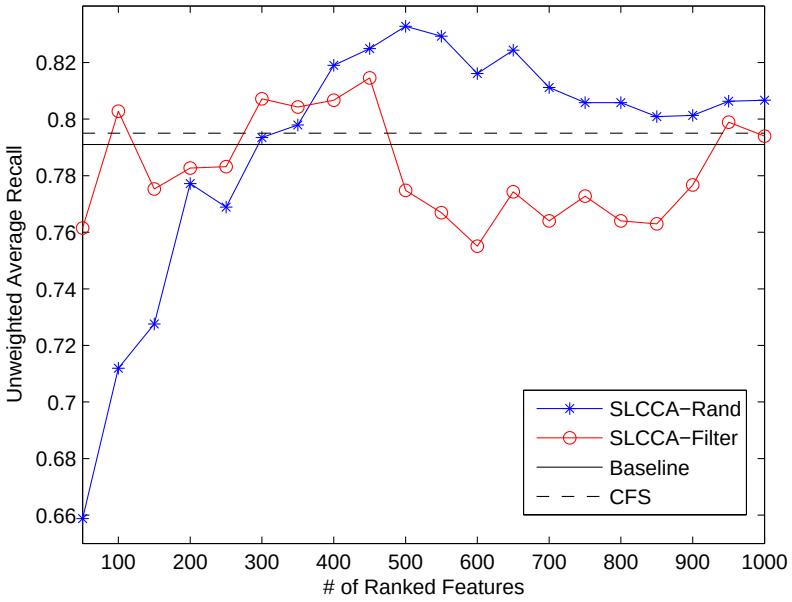

Figure 4.4. SLLCA-Rand, SLCCA-Filter and Baseline Comparison using Regression Labels for Feature Selection.

The experiments related to comparison of PCA versions of the methods are also investigated for Conflict data. Table 4.13 shows the results for SLLCA Filter and PCASLCCA Filter and table 4.14 for SLCCA Rand and PCASLCCA Rand, respectively. Different from Depression corpus, the experiment results indicate that PCA version of base methods is not successful for Conflict corpus.

We finally evaluate the proposed method on the challenge test set using the setting giving the best development set UAR performance. We restrict our test set trials to four: we use the first 500 features that yield the best development set results learned from the training set and the same number of features ranked by training and development set together both with the two best SVM complexity parameters. Using the features learned from the training set a test set UAR of 83.2% is reached. The UAR results improve to 84.2% when the proposed filter method is applied to the combined (training and development) set. The results achieved advance the state-of-the-art UAR (83.9%) on this corpus/protocol presented by the challenge winner [65].

Table 4.13. Comparison of SLCCA Filter and PCASLCCA Filter by using ELM Linear Kernel.

# of features	**SLCCA Filter**	**PCASLCCA Filter**
50	79.15	67.35
100	85.25	72.56
150	81.71	73.83
200	78.76	71.52
250	77.73	74.13
300	81.02	73.40
350	79.64	73.84
400	79.54	73.01
450	78.22	73.89
500	78.66	73.99
550	78.66	75.12
600	77.78	74.87
650	77.28	76.10
700	77.28	76.94
750	76.45	76.99
800	76.89	78.27
850	76.84	78.66
900	77.68	78.32
950	76.94	77.38
1000	77.33	77.82

Table 4.14. Comparison of SLCCA Rand and PCASLCCA Rand by using ELM Linear Kernel.

# of features	**SLCCA Rand**	**PCASLCCA Rand**
50	72.92	67.84
100	77.50	74.87
150	76.25	72.75
200	78.33	72.45
250	77.92	75.11
300	81.25	75.41
350	82.08	76.89
400	82.08	74.58
450	80.83	73.84
500	81.67	74.73
550	82.50	75.17
600	83.75	75.17
650	82.92	78.32
700	83.75	77.48
750	82.50	79.20
800	83.75	78.36
850	83.75	79.25
900	82.08	77.53
950	82.08	78.36
1000	82.08	78.36

5. CONCLUSION

In this thesis, a novel feature selection approach, based on a recently introduced discriminative projection based filter, is proposed. As a preliminary study not discussed in this thesis, the base feature selection method was applied to groups of features partitioned by the LLD information [33]. The success of that study on INTERSPEECH 2014 Physical-Load sub-challenge further motivated this study. While the preliminary work used domain-knowledge hints to divide-and-conquer the large feature set, the proposed method in this thesis does not necessitate domain knowledge. It uses the power of stochasticity to obtain feature subsets to avoid the curse of dimensionality. It overcomes the traps of local minimum commonly observed in greedy filter methods by learning feature level and feature group level weights in a variety of random contexts.

The efficacy of the proposed method is evaluated in two recent challenge corpora where the method is compared with challenge baseline, and two other benchmark methods. Correlation based Feature Selection [53] and the base method extended here, SLCCA-Filter [32] are used as benchmark methods. In both datasets, the proposed method performs better using the corresponding challenge competition measures Unweighted Average Recall (UAR) for classification and Root Mean Square Error (RMSE) for regression.

While the challenge in INTERSPEECH 2013 Conflict corpus was intended for binary classification, the baseline data provided also contains regression labels. We observe that learning ranking using regression labels yields better results than using class labels both in SLCCA-Filter and in SLCCA-Rand. The decreased observed performance in feature selection using class labels is attributed to loss of information during discretization. It is intuitive that regression labels have more information to drive correlation based feature selection. This point needs further investigation, and might be of valuable use in classification tasks where regression labels are available.

Apart from the increased performance in terms of classification/regression, utilizing a divide-and-conquer method is important to reduce the learning/memory complexity. First, the data need not be totally loaded into memory for feature selection. This might be an important advantage considering the shift to utilize BIG DATA in machine learning. Moreover, considering the $O(D^3)$ complexity of covariance inversion, the speed up is cubic when the feature set is partitioned into subsets.

In the final version of the proposed method, we split the feature set randomly into two and apply a discriminative projection (here CCA) to both partitions. Here, one of the parameters of the initial version is effectively eliminated. However, it is possible to introduce another parameter to the algorithm to control the number of partitions. The extension in this direction is left for future work.

The proposed method can further be investigated with other learners such as Artificial Neural Networks and Mixture Models. The choice of SVMs and Tree Bagging in this thesis was motivated by the speed of learning and comparability with the previous work on the same corpora.

The idea of feature selection from random subsets can be applied in other filter methods as well. However, in case where the filter method does not provide feature saliency, the aggregation of features can be at binary (e. g. set intersection/union) level rather than the elaborate approach use herein.

APPENDIX A: DETAILED RESULT TABLES

Table A.1. UAR (%) Performance of SLCCA-Rand Ranking Based on Regression Labels (D=3150, T=20).

#F/C	10^{-5}	10^{-4}	10^{-3}	0.01	0.1	1	10	100	Max
50	50.00	50.00	55.75	63.08	65.43	65.88	65.48	64.99	**65.9**
100	50.00	50.00	67.80	70.35	71.19	70.45	70.40	70.89	**71.2**
150	50.00	50.00	69.62	72.07	70.64	72.32	71.13	72.76	**72.8**
200	50.00	50.44	71.93	73.89	75.86	77.33	77.72	75.90	**77.7**
250	50.00	53.10	72.42	72.22	74.43	72.95	76.89	72.80	**76.9**
300	50.00	57.52	76.01	79.25	79.35	76.35	75.07	75.07	**79.3**
350	50.00	61.06	75.96	78.86	79.79	76.35	75.21	75.21	**79.8**
400	50.00	63.32	77.38	81.90	76.79	75.61	76.49	76.49	**81.9**
450	50.00	65.09	77.82	82.49	74.97	74.08	74.08	74.08	**82.5**
500	50.00	66.86	77.38	83.28	76.64	76.74	76.74	76.74	**83.3**
550	50.00	67.75	78.66	82.93	76.35	74.67	74.67	74.67	**82.9**
600	50.00	68.24	79.59	81.61	75.95	74.97	74.97	74.97	**81.6**
650	50.00	69.12	81.31	82.44	77.43	76.98	76.98	76.98	**82.4**
700	50.00	69.52	80.04	81.12	76.25	76.64	76.64	76.64	**81.1**
750	50.00	69.08	79.64	80.58	78.71	79.15	79.15	79.15	**80.6**
800	50.00	69.08	80.58	80.58	76.49	76.49	76.49	76.49	**80.6**
850	50.00	70.01	80.09	79.64	75.71	75.71	75.71	75.71	**80.1**
900	50.00	70.45	79.84	80.13	75.21	75.21	75.21	75.21	**80.1**
950	50.00	70.50	80.63	80.58	74.92	74.92	74.92	74.92	**80.6**
1000	50.00	71.78	80.67	80.63	74.03	74.03	74.03	74.03	**80.7**

Table A.2. UAR (%) Performance of SLCCA-Rand Ranking Based on Class Labels (D=3150, T=20). Columns represent varying SVM complexity parameter C, whereas rows correspond to varying number of ranked features (#F).

#F/C	$\mathbf{10^{-5}}$	$\mathbf{10^{-4}}$	$\mathbf{10^{-3}}$	**0.01**	**0.1**	**1**	**10**	**100**	**Max**
50	50.00	50.00	66.08	72.47	74.82	74.78	74.78	74.78	**74.82**
100	50.00	50.00	69.52	72.96	73.55	72.81	73.30	74.18	**74.18**
150	50.00	50.44	71.88	70.94	74.82	78.32	76.79	75.51	**78.32**
200	50.00	50.44	72.42	72.66	74.87	75.27	74.72	75.61	**75.61**
250	50.00	53.54	76.84	77.13	79.00	76.59	77.48	79.00	**79.00**
300	50.00	57.96	75.17	75.41	78.26	78.36	75.80	75.80	**78.36**
350	50.00	61.11	74.73	78.12	77.77	74.18	72.90	72.90	**78.12**
400	50.00	62.00	77.38	77.77	77.33	76.00	76.83	76.83	**77.77**
450	50.00	65.54	78.27	78.17	77.87	78.02	78.02	78.02	**78.27**
500	50.00	66.86	77.43	79.40	79.20	77.33	77.33	77.33	**79.40**
550	50.00	67.75	77.82	77.23	74.43	74.92	74.92	74.92	**77.82**
600	50.00	67.75	76.55	79.84	77.87	76.29	76.29	76.29	**79.84**
650	50.00	69.08	77.33	80.28	77.52	75.46	75.46	75.46	**80.28**
700	50.00	69.08	77.43	80.63	75.41	74.57	74.57	74.57	**80.63**
750	50.00	69.57	79.20	80.58	77.18	77.62	77.62	77.62	**80.58**
800	50.00	69.17	79.69	80.18	77.52	76.74	76.74	76.74	**80.18**
850	50.00	70.01	79.64	79.25	75.41	76.64	76.64	76.64	**79.64**
900	50.00	70.99	80.18	80.53	79.49	79.49	79.49	79.49	**80.53**
950	50.00	70.99	80.97	79.79	77.87	77.87	77.87	77.87	**80.97**
1000	50.00	71.43	80.67	78.95	77.03	77.03	77.03	77.03	**80.67**

Table A.3. UAR (%) Performance of SLCCA-Filter Ranking Based on Regression Labels (D=3150, T=20). Columns represent varying SVM complexity parameter C, whereas rows correspond to varying number of ranked features (#F).

#F/C	$\mathbf{10^{-5}}$	$\mathbf{10^{-4}}$	$\mathbf{10^{-3}}$	**0.01**	**0.1**	**1**	**10**	**100**	**Max**
50	50.00	50.00	67.31	74.78	76.15	76.15	76.15	75.32	**76.15**
100	50.00	50.00	74.43	78.86	80.28	76.99	76.94	75.71	**80.28**
150	50.00	50.44	74.83	77.53	75.86	73.84	74.78	75.12	**77.53**
200	50.00	52.21	76.35	78.27	77.67	76.40	75.51	75.26	**78.27**
250	50.00	53.54	74.48	78.32	72.56	73.79	71.57	72.75	**78.32**
300	50.00	58.46	76.50	80.72	74.72	76.00	72.55	71.77	**80.72**
350	50.00	61.11	77.3	80.4	75.21	73.58	72.85	72.85	**80.43**
400	50.00	62.44	76.94	80.67	77.33	72.36	74.08	74.08	**80.67**
450	50.00	65.09	78.66	81.46	75.60	73.54	73.54	73.54	**81.46**
500	50.00	65.98	76.40	77.48	73.88	73.24	73.24	73.24	**77.48**
550	50.00	66.81	75.96	76.69	75.51	74.72	74.72	74.72	**76.69**
600	50.00	67.26	75.51	75.32	74.57	75.51	75.51	75.51	**75.51**
650	50.00	67.26	75.51	77.43	72.80	74.62	74.62	74.62	**77.43**
700	50.00	68.14	76.40	75.36	74.97	72.46	72.46	72.46	**76.40**
750	50.00	68.58	77.28	76.40	74.23	74.23	74.23	74.23	**77.28**
800	50.00	67.70	76.40	75.86	75.85	75.85	75.85	75.85	**76.40**
850	50.00	66.81	74.58	76.30	75.26	75.26	75.26	75.26	**76.30**
900	50.00	67.70	75.02	77.67	76.69	76.69	76.69	76.69	**77.67**
950	50.00	66.81	75.07	76.35	79.89	79.89	79.89	79.89	**79.89**
1000	50.00	65.93	76.35	77.58	79.40	79.40	79.40	79.40	**79.40**

Table A.4. UAR (%) Performance of SLCCA-Filter Ranking Based on Class Labels (D=3150, T=20). Columns represent varying SVM complexity parameter C, whereas rows correspond to varying number of ranked features (#F).

#F/C	**10^{-5}**	**10^{-4}**	**10^{-3}**	**0.01**	**0.1**	**1**	**10**	**100**	**Max**
50	50	50	64.31	74.28	76.10	74.97	73.99	75.27	**76.10**
100	50.00	50.00	74.39	79.99	76.40	74.72	74.23	77.13	**79.99**
150	50.00	50.44	73.25	79.50	77.53	77.18	75.07	75.81	**79.50**
200	50.00	50.44	76.25	77.92	74.63	75.17	70.69	66.71	**77.92**
250	50.00	52.21	77.19	77.33	75.21	69.90	69.61	70.69	**77.33**
300	50.00	55.80	77.19	77.92	74.13	69.60	66.61	67.00	**77.92**
350	50.00	59.78	75.56	76.89	74.52	70.39	69.95	69.95	**76.89**
400	50.00	61.55	76.35	76.10	75.21	70.83	70.00	70.00	**76.35**
450	50.00	64.21	75.96	76.59	74.77	73.25	73.25	73.25	**76.59**
500	50.00	65.54	77.28	76.20	73.54	75.21	75.21	75.21	**77.28**
550	50.00	66.37	76.40	75.27	73.98	73.89	73.89	73.89	**76.40**
600	50.00	67.26	75.51	76.64	74.62	72.90	72.90	72.90	**76.64**
650	50.00	67.26	75.51	76.99	74.92	75.36	75.36	75.36	**76.99**
700	50.00	68.58	76.84	76.79	73.79	72.95	72.95	72.95	**76.84**
750	50.00	68.58	77.73	77.28	75.90	75.90	75.90	75.90	**77.73**
800	50.00	67.70	76.40	75.86	75.85	75.85	75.85	75.85	**76.40**
850	50.00	66.81	74.58	76.30	75.26	75.26	75.26	75.26	**76.30**
900	50.00	67.70	75.02	77.67	76.69	76.69	76.69	76.69	**77.67**
950	50.00	66.81	75.07	76.35	79.89	79.89	79.89	79.89	**79.89**
1000	50.00	65.93	76.35	77.58	79.40	79.40	79.40	79.40	**79.40**

REFERENCES

1. Schuller, B., S. Steidl, A. Batliner, F. Burkhardt, L. Devillers, C. Müller and S. Narayanan, "Paralinguistics in Speech and Language—State-of-the-art and The Challenge", *Computer Speech and Language*, Vol. 27, No. 1, pp. 4 – 39, 2013.

2. Schuller, B., M. Wöllmer, F. Eyben and G. Rigoll, "Retrieval of Paralinguistic Information in Broadcasts", M. T. Maybury (Editor), *Multimedia Information Extraction: Advances in Video, Audio, and Imagery Extraction for Search, Data Mining, Surveillance, and Authoring*, chap. 17, pp. 273–288, Wiley, IEEE Computer Society Press, New Jersey, 2011.

3. Martinez, C. and A. Cruz, "Emotion Recognition in Non-structured Utterances for Human-Robot Interaction", *Robot and Human Interactive Communication, 2005. ROMAN 2005. IEEE International Workshop on*, pp. 19–23, 2005.

4. Batliner, A., S. Steidl and E. Noth, "Associating Children's Non-verbal and Verbal Behaviour: Body movements,Emotions, and Laughter in a Human-Robot Interaction", *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, pp. 5828–5831, 2011.

5. Delaborde, A. and L. Devillers, "Use of Nonverbal Speech Cues in Social Interaction Between Human and Robot: Emotional and Interactional Markers", *Proceedings of the 3rd International Workshop on Affective Interaction in Natural Environments*, AFFINE '10, pp. 75–80, ACM, New York, NY, USA, 2010.

6. Schroder, M., R. Cowie, D. K. J. Heylen, M. Pantic, C. Pelachaud and B. Schuller, "Towards Responsive Sensitive Artificial Listeners", *Proceedings of the Fourth International Workshop on Human-Computer Conversation*, University of Sheffield, Sheffield, UK, 2008.

7. Belin, P., S. Fillion-Bilodeau and F. Gosselin, "The Montreal Affective Voices:

A Validated Set of Nonverbal Affect Bursts for Research on Auditory Affective Processing", *Behavior Research Methods*, Vol. 40, No. 2, pp. 531–539, 2008.

8. Schoentgen, J., "Vocal Cues of Disordered Voices: An Overview", *Acta Acustica united with Acustica*, Vol. 92, No. 5, pp. 667–680, 2006.

9. Rektorova, I., J. Barrett, M. Mikl, I. Rektor and T. Paus, "Functional Abnormalities in The Primary or Facial Sensorimotor Cortex during Speech in Parkinson's Disease", *Movement Disorders*, Vol. 22, No. 14, pp. 2043–2051, 2007.

10. Maier, A., T. Haderlein, U. Eysholdt, F. Rosanowski, A. Batliner, M. Schuster and E. Nöth, "{PEAKS} – A System for The Automatic Evaluation of Voice and Speech Disorders", *Speech Communication*, Vol. 51, No. 5, pp. 425 – 437, 2009.

11. Malyska, N., T. F. Quatieri and D. Sturim, "Automatic Dysphonia Recognition using Biologically Inspired Amplitude-Modulation Features", *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2005)*, pp. 873–876, 2005.

12. Dibazar, A., S. Narayanan and T. Berger, "Feature Analysis for Automatic Detection of Pathological Speech", *Engineering in Medicine and Biology, 2002. 24th Annual Conference and the Annual Fall Meeting of the Biomedical Engineering Society EMBS/BMES Conference, 2002. Proceedings of the Second Joint*, Vol. 1, pp. 182–183 vol.1, 2002.

13. Schuller, B., S. Steidl, A. Batliner, A. Vinciarelli, K. Scherer, F. Ringeval, M. Chetouani, F. Weninger, F. Eyben, E. Marchi, M. Mortillaro, H. Salamin, A. Polychroniou, F. Valente and S. Kim, "The INTERSPEECH 2013 Computational Paralinguistics Challenge: Social Signals, Conflict, Emotion, Autism", *Proceedings of the INTERSPEECH*, pp. 148–152, ISCA, Lyon, France, 2013.

14. Valstar, M., B. Schuller, K. Smith, F. Eyben, B. Jiang, S. Bilakhia, S. Schnieder, R. Cowie and M. Pantic, "AVEC 2013–The Continuous Audio/Visual Emotion

and Depression Recognition Challenge", *Proceeding of the 3rd ACM International Workshop on Audio/Visual Emotion Challenge*, AVEC '13, pp. 3–10, 2013.

15. Batliner, A., B. Schuller, S. Schaeffler and S. Steidl, "Mothers, Adults, Children, Pets; Towards The Acoustics of Intimacy", *Proceedings IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2008)*, pp. 4497–4500, 2008.

16. Enos, F., E. Shriberg, M. Graciarena, J. Hirschberg and A. Stolcke, "Detecting Deception using Critical Segments", *Proceedings of the INTERSPEECH*, pp. 2281–2284, 2007.

17. Zhao, X., S. Zhang and B. Lei, "Robust Emotion Recognition in Noisy Speech via Sparse Representation", *Neural Computing and Applications*, Vol. 24, No. 7-8, pp. 1539–1553, 2014.

18. Schuller, B., R. Müller, F. Eyben, J. Gast, B. Hörnler, M. Wöllmer, G. Rigoll, A. Höthker and H. Konosu, "Being Bored? Recognising Natural Interest by Extensive Audiovisual Integration for Real-life Application", *Image Vision Computing*, Vol. 27, No. 12, pp. 1760–1774, 2009.

19. Schiel, F. and C. Heinrich, "Laying the Foundation for In-car Alcohol Detection by Speech", M. Uther (Editor), *Proceedings of the INTERSPEECH*, pp. 983–986, 2009.

20. Krajewski, J. and B. J. Kröger, "Using Prosodic and Spectral Characteristics for Sleepiness Detection", *Proceedings of the INTERSPEECH*, pp. 1841–1844, 2007.

21. Haderlein, T., E. Nöth, H. Toy, A. Batliner, M. Schuster, U. Eysholdt, J. Hornegger and F. Rosanowski, "Automatic Evaluation of Prosodic Features of Tracheoesophageal Substitute Voice", *European Archives of Oto-Rhino-Laryngology*, Vol. 264, No. 11, pp. 1315–1321, 2007.

22. Hansen, J. H., S. E. Bou-Ghazale, R. Sarikaya and B. Pellom, "Getting Started with SUSAS: A Speech under Simulated and Actual Stress Database", Vol. 97, pp. 1743–46, 1997.

23. Mporas, I. and T. Ganchev, "Estimation of Unknown Speaker's Height from Speech", *International Journal of Speech Technology*, Vol. 12, No. 4, pp. 149–160, 2009.

24. Weiss, B. and F. Burkhardt, "Voice Attributes Affecting Likability Perception", *Proceedings of the INTERSPEECH*, pp. 2014–2017, 2010.

25. Russell, J. A., J.-A. Bachorowski and J.-M. Fernández-Dols, "Facial and Vocal Expressions of Emotion", *Annual Review of Psychology*, Vol. 54, No. 1, pp. 329–349, 2003.

26. Campbell, N., H. Kashioka and R. Ohara, "No Laughing Matter", *Proceedings of the INTERSPEECH*, pp. 465–468, 2005.

27. Pal, P., A. Iyer and R. Yantorno, "Emotion Detection from Infant Facial Expressions and Cries", *Proceedings IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2006)*, Vol. 2, pp. II–II, 2006.

28. Matos, S., S. Birring, I. Pavord and D. Evans, "Detection of Cough Signals in Continuous Audio Recordings using Hidden Markov Models", *Biomedical Engineering, IEEE Transactions on*, Vol. 53, No. 6, pp. 1078–1083, 2006.

29. Shuller, B., "Voice and Speech Analysis in Search of States and Traits", A. A. Salah and T. Gevers (Editors), *Computer Analysis of Human Behavior*, chap. 9, pp. 227–253, Springer London, 2011.

30. Schuller, B., S. Steidl and A. Batliner, "The Interspeech 2009 Emotion Challenge", *Proceedings of the INTERSPEECH*, pp. 312–315, ISCA, Brighton, UK, 2009.

31. Eyben, F., M. Wöllmer and B. Schuller, "Opensmile: The Munich Versatile and Fast Open-source Audio Feature Extractor", *Proceeding of the International Conference on Multimedia*, pp. 1459–1462, ACM, 2010.

32. Kaya, H., F. Eyben, A. A. Salah and B. W. Schuller, "CCA Based Feature Selection with Application to Continuous Depression Recognition from Acoustic Speech Features", *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2014)*, pp. 3757–3761, 2014.

33. Kaya, H., T. Özkaptan, A. A. Salah and S. F. Gürgen, "Canonical Correlation Analysis and Local Fisher Discriminant Analysis based Multi-View Acoustic Feature Reduction for Physical Load Prediction", *Proceedings of the INTERSPEECH*, ISCA, Singapore, 2014.

34. Hotelling, H., "Relations Between Two Sets of Variates", *Biometrika*, Vol. 28, No. 3/4, pp. 321–377, 1936.

35. Hardoon, D. R., S. Szedmak and J. Shawe-Taylor, "Canonical Correlation Analysis: An Overview with Application to Learning Methods", *Neural Computation*, Vol. 16, No. 12, pp. 2639–2664, 2004.

36. Andrew, G., R. Arora, J. Bilmes and K. Livescu, "Deep Canonical Correlation Analysis", *Proceeding of the 30th International Conference on Machine Learning*, pp. 1247–1255, Atlanta ,Georgia, USA, 2013.

37. Fukunaga, K., *Introduction to Statistical Pattern Recognition*, Academic Press, Boston, 2nd edn., 1990.

38. He, X. and P. Niyogi, "Locality Preserving Projections", MIT Press, Cambridge, MA, 2004.

39. Sugiyama, M., "Local Fisher Discriminant Analysis for Supervised Dimensionality Reduction", *Proceeding of the 23rd International Conference on Machine Learning*,

ICML '06, pp. 905–912, ACM, New York, NY, USA, 2006.

40. Huang, G.-B., Q.-Y. Zhu and C.-K. Siew, "Extreme Learning Machine: A New Learning Scheme of Feedforward Neural Networks", *Neural Networks, 2004. Proceedings. 2004 IEEE International Joint Conference on*, Vol. 2, pp. 985–990, IEEE, 2004.

41. Huang, G.-B., Q.-Y. Zhu and C.-K. Siew, "Extreme Learning Machine: Theory and Applications", *Neurocomputing*, Vol. 70, No. 1, pp. 489–501, 2006.

42. Suykens, J. A. and J. Vandewalle, "Least Squares Support Vector Machine Classifiers", *Neural Processing Letters*, Vol. 9, No. 3, pp. 293–300, 1999.

43. Huang, G.-B., H. Zhou, X. Ding and R. Zhang, "Extreme Learning Machine for Regression and Multiclass Classification", *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, Vol. 42, No. 2, pp. 513–529, 2012.

44. Huang, G.-B., D. H. Wang and Y. Lan, "Extreme Learning Machines: A Survey", *International Journal of Machine Learning and Cybernetics*, Vol. 2, No. 2, pp. 107–122, 2011.

45. Cambria, E., G.-B. Huang, L. L. C. Kasun, H. Zhou, C.-M. Vong, J. Lin, J. Yin, Z. Cai, Q. Liu, K. Li *et al.*, "Extreme Learning Machines", *IEEE Intelligent Systems*, Vol. 28, No. 6, pp. 30–59, 2013.

46. Rao, C. R. and S. K. Mitra, *Generalized Inverse of Matrices and Its Applications*, Vol. 7, Wiley New York, 1971.

47. Bartlett, P. L., "The Sample Complexity of Pattern Classification with Neural Networks: The Size of the Weights is More Important than The Size of the Network", *Information Theory, IEEE Transactions on*, Vol. 44, No. 2, pp. 525–536, 1998.

48. Blum, A. L. and P. Langley, "Selection of Relevant Features and Examples in

Machine Learning", *Artificial Intelligence*, Vol. 97, pp. 245–271, 1997.

49. Reunanen, J., "Overfitting in Making Comparisons Between Variable Selection Methods", *Journal of Machine Learning Research*, Vol. 3, pp. 1371–1382, 2003.

50. Marill, T. and D. M. Green, "On The Effectiveness of Receptors in Recognition Systems", *Information Theory, IEEE Transactions on*, Vol. 9, No. 1, pp. 11–17, 1963.

51. Whitney, A. W., "A Direct Method of Nonparametric Measurement Selection", *Computers, IEEE Transactions on*, Vol. 100, No. 9, pp. 1100–1103, 1971.

52. Pudil, P., J. Novovičová and J. Kittler, "Floating Search Methods in Feature Selection", *Pattern Recognition Letters*, Vol. 15, No. 11, pp. 1119–1125, 1994.

53. Hall, M. A., *Correlation-based Feature Selection for Machine Learning*, Ph.D. Thesis, The University of Waikato, 1999.

54. Peng, H., F. Long and C. Ding, "Feature Selection Based on Mutual Information: Criteria of Max-Dependency, Max-Relevance, and Min-Redundancy", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 27, No. 8, pp. 1226–1238, 2005.

55. Quinlan, J. R., "Induction of Decision Trees", *Machine Learning*, pp. 81–106, 1986.

56. Shannon, C. E., "A Mathematical Theory of Communication", *Bell System Technical Journal*, Vol. 27, No. 3, pp. 379–423, 1948.

57. Press, W. H., S. A. Teukolsky, W. T. Vetterling and B. P. Flannery, *Numerical Recipes in C (2Nd Ed.): The Art of Scientific Computing*, Cambridge University Press, New York, NY, USA, 1992.

58. Sakar, C. O., O. Kursun and F. Gürgen, "A Feature Selection Method Based on Kernel Canonical Correlation Analysis and The Minimum Redundancy Maximum

Relevance Filter Method", *Expert Systems with Applications*, Vol. 39, No. 3, pp. 3432–3437, 2012.

59. Asgari, M., A. Bayestehtashk and I. Shafran, "Robust and Accurate Features for Detecting and Diagnosing Autism Spectrum Disorders", F. Bimbot, C. Cerisara, C. Fougeron, G. Gravier, L. Lamel, F. Pellegrino and P. Perrier (Editors), *Proceedings of the INTERSPEECH*, pp. 191–194, ISCA, 2013.

60. Torres, J., A. Saad and E. Moore, "Evaluation of Objective Features for Classification of Clinical Depression in Speech by Genetic Programming. Accepted for Publication", *11th Online World Conference on Soft Computing in Industrial Applications*, 2006.

61. Park, C.-H. and K.-B. Sim, "The Novel Feature Selection Method Based on Emotion Recognition System", D.-S. Huang, K. Li and G. Irwin (Editors), *Computational Intelligence and Bioinformatics*, Vol. 4115 of *Lecture Notes in Computer Science*, pp. 731–740, Springer Berlin Heidelberg, 2006.

62. Torres, J., A. Saad and E. Moore, "Application of A GA/Bayesian Filter-Wrapper Feature Selection Method to Classification of Clinical Depression from Speech Data", A. Saad, K. Dahal, M. Sarfraz and R. Roy (Editors), *Soft Computing in Industrial Applications*, Vol. 39 of *Advances in Soft Computing*, pp. 115–121, Springer Berlin Heidelberg, 2007.

63. Espinosa, H., J. Garcia and L. Pineda, "Bilingual Acoustic Feature Selection for Emotion Estimation using A 3D Continuous Model", *Automatic Face Gesture Recognition and Workshops (FG 2011), 2011 IEEE International Conference on*, pp. 786–791, 2011.

64. Giannoulis, P. and G. Potamianos, "A Hierarchical Approach with Feature Selection for Emotion Recognition from Speech", N. C. C. Chair), K. Choukri, T. Declerck, M. U. Doğan, B. Maegaard, J. Mariani, A. Moreno, J. Odijk and S. Piperidis (Editors), *Proceedings of the Eight International Conference on Lan-*

guage Resources and Evaluation (LREC'12), European Language Resources Association (ELRA), Istanbul, Turkey, 2012.

65. Räsänen, O. and J. Pohjalainen, "Random Subset Feature Selection in Automatic Recognition of Developmental Disorders, Affective States, and Level of Conflict from Speech.", F. Bimbot, C. Cerisara, C. Fougeron, G. Gravier, L. Lamel, F. Pellegrino and P. Perrier (Editors), *Proceedings of the INTERSPEECH*, pp. 210–214, ISCA, 2013.

66. Kirchhoff, K., Y. Liu and J. Bilmes, "Classification of Developmental Disorders from Speech Signals using Submodular Feature Selection.", F. Bimbot, C. Cerisara, C. Fougeron, G. Gravier, L. Lamel, F. Pellegrino and P. Perrier (Editors), *Proceedings of the INTERSPEECH*, pp. 187–190, ISCA, 2013.

67. Moore, J., L. Tian and C. Lai, "Word-Level Emotion Recognition Using High-Level Features", A. Gelbukh (Editor), *Computational Linguistics and Intelligent Text Processing*, Vol. 8404 of *Lecture Notes in Computer Science*, pp. 17–31, Springer Berlin Heidelberg, 2014.

68. Bejani, M., D. Gharavian and N. Charkari, "Audiovisual Emotion Recognition using ANOVA Feature Selection Method and Multi-classifier Neural Networks", *Neural Computing and Applications*, Vol. 24, No. 2, pp. 399–412, 2014.

69. Kim, S., M. Filippone, F. Valente and A. Vinciarelli, "Predicting Continuous Conflict Perception with Bayesian Gaussian Processes", *IEEE Transactions on Affective Computing (to appear)*, 2014.

70. Pudil, P., J. Novovičová and J. Kittler, "Floating Search Methods in Feature Selection", *Pattern Recognition Letters*, Vol. 15, No. 11, pp. 1119 – 1125, 1994.

71. Beck, A., R. Steer, R. Ball and W. Ranieri, "Comparison of Beck Depression Inventories -IA and -II in Psychiatric Outpatients", *Journal of Personality Assessment*, Vol. 67, No. 3, pp. 588–597, 1996.

72. Hall, M., E. Frank, G. Holmes, B. Pfahringer, P. Reutemann and I. H. Witten, "The WEKA Data Mining Software: An Update", *SIGKDD Explorations Newsletter*, Vol. 11, No. 1, pp. 10–18, 2009.

73. Kim, S., M. Filippone, F. Valente and A. Vinciarelli, "Predicting The Conflict Level in Television Political Debates: An Approach Based on Crowdsourcing, Nonverbal Communication and Gaussian Processes", *Proceedings of the 20th ACM international conference on Multimedia*, pp. 793–796, ACM, 2012.

74. Vinciarelli, A., S. Kim, F. Valente and H. Salamin, "Collecting Data for Socially Intelligent Surveillance and Monitoring Approaches: The Case of Conflict in Competitive Conversations", *International Symposium on Communications, Control, and Signal Processing*, 2012.

75. Schuller, B., S. Steidl, A. Batliner, E. Nöth, A. Vinciarelli, F. Burkhardt, R. van Son, F. Weninger, F. Eyben, T. Bocklet *et al.*, "The INTERSPEECH 2012 Speaker Trait Challenge.", *Proceedings of the INTERSPEECH*, 2012.