Interpretability in Multidimensional Classification

Vincent Vanhoucke[1,2] and Rosaria Silipo[2]

[1] Stanford University, Stanford CA 94305, USA
[2] Nuance Communications, 1380 Willow Road, Menlo Park CA 94025, USA

Abstract. Generating rule-based models from data is an efficient way of inferring information from large datasets. In high dimensional spaces, the complexity of the model itself can undermine the direct interpretability of this information. This chapter introduces metrics quantifying the information flow between inputs, feature dimensions and output classes. These metrics are used to estimate the contribution of individual input features to a fuzzy classification task without making explicit use of the data underlying the model. Application of these techniques to a speech classification problem shows that significant reduction in the model dimensionality can be achieved with minimal accuracy loss.

1 Introduction

1.1 Classification Algorithms and Interpretability

The increasing accessibility of data in many areas of technology has driven the development of many automatic classification and clustering algorithms. The use of these algorithms for classification and modeling purposes is in general well understood. However, the more complex these models are, the more they tend to obfuscate the relationships between the data and the classification output.

A more recent research trend focuses on investigating the data with the goal of getting some insights about the system that generated it. A classification algorithm is now required to provide a low error rate as well as an interpretable decision process.

The task of classification can be considered, in its canonical form, as the application of a one-to-one mapping from an input feature space into the space of output classes. In addition, it is of importance both when designing and using a classifier to be able to evaluate it on several grounds. Among these:

- The classifier's confidence in its decisions
- The interpretability of the relationships between inputs and outputs
- The sensitivity to the input features

The first automatic classifiers that tried to answer the need for interpretability were based on fuzzy logic. Fuzzy logic based classifiers have been

introduced to facilitate the interpretation process [1,2] [3, Chapter 8]. The representation of knowledge through rules and of the input space through linguistic values allows the user to easily understand how a given output class has been assigned to a given input pattern.

For input spaces with small dimensionality, fuzzy rules are easy to read. If the input space includes many features, or if the problem is complex and a very high number of fuzzy rules is generated, the decision process becomes much less transparent.

Owing to the dramatic growth of the dimensionality of databases, even as interpretable classifiers as fuzzy models tend to show their limits. It is sometimes necessary to set aside the rule-based representation, and move to a more compact description of the system based on information content. A more global measure of the system sensitivity to the input features can provide better insights into the classifier [4,5], using a much smaller set of descriptive variables.

Statistical decision trees, based on probabilistic observations, were also introduced to provide an interpretable statistical classification process [6,7]. In statistical decision trees, at each step, the entropy maximization on a given subset of training data is used to determine the most informative split of the input space on one of the input dimensions. After recursively applying such split search, a tree can be built where the nodes correspond to the decisions and the leaves to the final classification results. A visual inspection of the tree can rapidly provide a summary about the whole classification process. Given that statistical decision trees and fuzzy systems represent two among the most interpretable classification algorithms, they were combined to produce fuzzy decision trees (fuzzy ID3) [9]. On the basis of a fuzzy entropy definition, a fuzzy decision tree can be built in a similar way to statistical decision trees. However, even with these tools, whenever the classification task is complex and/or the input space has very high dimensionality, the visual inspection of the model can become intractable. In addition, there is a need for quantifiable measures to be associated with the interpretation process. A quantitative "interpreter" associated with a classification model can be much more powerful a tool for automated system analysis than a mechanism requiring visual inspection.

1.2 Challenging Datasets

Large and high dimensional databases are increasingly becoming available to the community to challenge the way data analysis has been performed so far.

An example of a field that produces large size databases with high dimensionality is speech recognition. Research speech recognition systems compute up to 200 input features from a single frame of the speech signal. The OGI Corpus [10], for example, consists of one minute long speech segments spoken over commercial telephone lines of speakers in 12 different languages. The database contains a total of more than 1900 calls. Numerous corpora of

similar or greater size are quite widespread and intensively exploited to train and analyze speech systems.

In the biomedical field, the recording of heterogeneous data is becoming more widespread. For example, for the electrocardiographic signal (ECG) the number of leads increased (up to 12) as well as the duration of each recording (up to 24 hours). Many physiological sources are now monitored simultaneously for longer and longer periods of time.

Many examples of this kind of biomedical data can be found on the PhysioBank web site [11]. PhysioBank is a large and growing archive of well-characterized digital recordings of physiologic signals and related data for use by the biomedical research community. PhysioBank currently includes databases of multi-parameter cardiopulmonary, neural, and other biomedical signals from healthy subjects and patients with a variety of conditions with major public health implications, including sudden cardiac death, congestive heart failure, epilepsy, gait disorders, sleep apnea, and aging. For example, the Apnea-ECG database consists of 70 ECG recordings with simultaneous respiration signals, each typically 8 hours long [12].

Similarly, bioinformatics work almost exclusively on extremely large databases. The National Center for Biotechnology Information (NCBI) [13] creates public databases on computational biology, genome data, and biomedical information, all of these aimed at the better understanding of molecular processes affecting human health and diseases [14].

In particular, GenBank [15] is the NIH genetic sequence database, an annotated collection of all publicly available DNA sequences. GenBank (at NCBI), together with the DNA DataBank of Japan (DDBJ) and the European Molecular Biology Laboratory (EMBL) comprise the International Nucleotide Sequence Database Collaboration. These three organizations exchange data on a daily basis. GenBank grows at an exponential rate, with the number of nucleotide bases doubling approximately every 14 months. Currently, GenBank contains more than 13 billion bases from over 100,000 species.

1.3 Information Measures

On large and/or high-dimensional databases, direct interpretation techniques, such as visual inspection of decision trees and representation of fuzzy rules on a two or three-dimensional space, are not informative. Alternative techniques have to be implemented to gain insights on the most significant aspects of the classification system.

One approach has been to view the system in a parallel coordinate environment. All input space coordinates are represented in different sections of a two-dimensional plane, allowing an unconventional but exhaustive view of the classification system [16–18]. Other techniques focus on the definition of a global measure characterizing one or more properties of the system.

An important part of a decision process consists of the measurement of how much each input feature contributes to the final outcome. Feature characterization can be used for several purposes:

- feature selection: removing superfluous inputs to simplify classification,
- robustness evaluation: analyzing the classification outcome under noisy conditions,
- feature weighting: rebalancing of the feature importance in the decision process.

Many information retrieval or data mining techniques, developed to allow some partial interpretation of the decision process, operate directly on the data [4], independently from the classification model. However, there are also reasons to treat classification and interpretation jointly.

First, techniques that operate directly on the data often actually create a model of it. A model trained for the purposes of classification will guarantee a better match to the problem at hand. Secondly, the classifier has in general orders of magnitude less degrees of freedom than the data, making the analysis simpler and more tractable computationally. As a consequence, techniques involving joint modeling and classification are getting a lot of attention [19,20].

Fuzzy systems, in particular, produce a computationally simple description of the input space and can be considered good candidates for joint classification and interpretation analysis. In Section 3, a joint classification and feature selection approach is proposed. Fuzzy rules are trained and subsequently analyzed in terms of the impact that input features have on the classification process.

To quantify the information content associated with the input features of a fuzzy model, a mutual information measure is applied (Section 2.5), being the relative difference between the intrinsic information available in the system before and after using a given input feature for the analysis [21] [3, chapter 8]. The measure of the information contained in a fuzzy model is derived solely on the basis of its fuzzy rules.

For the purposes of feature selection, the input features are ranked according to their impact on the fuzzy classification model and the least influencing features are discarded. The ranking provides useful insights on the relative importance of the input features for this fuzzy model. The feature selection also reduces the model size and possibly improves its performance, since noisy input features that do not contribute to or even confuse the classification process are identified and excluded from the analysis.

1.4 Feature Selection

Given a large dimensional input space and a classifier, the objective of feature selection [22] is to determine from the model which sets of features are

meaningful to use and which can be discarded. There are several motivations for addressing this problem [23]:

- Reduce the number of rules to improve the interpretability of the model,
- Improve the accuracy of the model by removing potentially noisy superfluous features,
- Compress the model for complexity reduction and faster classification.

Features selection methods are usually referred to as either *filter* methods or *wrapper* methods. Filter methods quantify the importance of input features without taking into account the classification algorithm that will use the selected subset. Wrapper methods on the opposite use the classification algorithm as the evaluation function to choose the relevant input features. The evaluation functions, in general, can be distance, information (or uncertainty), dependence consistency, and classifier error rate [24].

Numerous criteria [25] have been proposed for selecting features, including correlation methods [26] and minimum description length [27]. A very extensive bibliography on feature selection can be found at [28].

In [21], for example, feature merit measures are defined on the basis of the entropy maximization theory of statistical decision trees. Another algorithm uses the ROC curves of the inductive algorithm to quantify the importance of the input features [29].

The main issue of feature selection is the intractability of an exhaustive search of the space of all possible subsets of features for high-dimensional problems. Various search strategies can be adopted [30,24], like sequential selection [31,32], or stochastic search strategies [33–35]. A general drawback of these techniques, however, is the amount of computation they require. These can involve a high number of training runs for different classifier configurations as well as decisions based on direct inspection of potentially large volumes of data.

1.5 Speech Classification

Automatic Speech Recognition (ASR) systems are classifiers that make use of various levels of linguistic information to decode spoken utterances into their textual transcription. Typical speech recognition systems consist of several data analysis blocks:

- Front-end: the speech signal is segmented into frames long of a few milliseconds, and passed trough an analysis transform from which features are extracted.
- Classifier (Acoustic Model): the features are submitted to a classifier that computes the probabilities of possible phonetic units.
- Decoder (Language Model): translates the probabilistic sequences of phonetic units into the most likely corresponding word.

ASR systems are usually very complex and involve a very high number of parameters derived from acoustic, phonetic and linguistic knowledge [36]. This structure usually does not provide much visibility into the workings of the system and the interpretability of ASR models is quite poor. As an example, any modification made to the front-end processing typically requires a retraining of the acoustic model, which can be an extremely time consuming process.

In commercial systems, typically between 20 and 50 spectral or time-domain features are extracted from the speech signal and used as input for the frame classification [36]. Due to the complexity of the system, it is rather difficult to estimate a priori the contribution of each input feature to the final result. For the same reason, any recursive search strategy aimed at reducing the dimensionality of the input space would involve a prohibitively large number of re-training and re-evaluation of the system performance.

The feature selection algorithm proposed in this chapter facilitates the task of investigating the input feature contributions. Because it uses fuzzy systems, and because it operates directly on the model while making abstraction of the data, it is computationally inexpensive and particularly suited for this class of problems.

The analysis in Section 4 focuses essentially on the influence of features on the acoustic model, with the assumption that a good frame classifier is the basis for an acceptable word recognition error rate. In particular, the phone classification task is decomposed into the recognition of simpler phonetic properties [37,38] from which phonetic units derive. This approach is more robust than direct classification of phones using a single all-purpose classifier. The phone labels can be recovered as a second step in the process by cross-analysis of all the phonetic properties of the speech frame [39].

2 Information Measures

2.1 Membership Degree and Information

A broad class of commonly used classifiers operate using mechanisms much more suitable for introspection than the simple specification of a functional mapping between inputs and outputs. These classifiers operate by defining a set of membership functions that associate a given input pattern from domain D to an output class $C \in \mathcal{C}$ by means of a membership degree $\mu_C : D \rightarrow [0, 1]$. The membership degree summarizes by one scalar the information the model possesses about the class membership of a given input vector, with the assumption that the membership degree is proportional to the model's confidence in the association.

Classifying an input $\boldsymbol{x}$ amounts to finding the class that maximizes the membership degree:

$$C(\boldsymbol{x}) = \arg\max_{\mathcal{C}} \mu_C(\boldsymbol{x})$$

This simple representation of the information contained in the classifier adds a lot to its interpretability. This type of model contains information about all the possible class associations for any given input vector, as opposed to the output class only. A mathematically simple representation of the decision process is thus embedded into the model, and can be exploited for introspective analysis.

A consequence of this choice of a representation is that comparing relative membership degrees across regions of the feature and/or output space will provide measures of the amount of information contained in the model relative to any combination of inputs and outputs. For example, the information contained in the model about any given input $\boldsymbol{x}$ is determined by the distribution of the membership degree of $\boldsymbol{x}$ across all possible classes. A very uneven distribution indicates that the model makes strong associations between this input and some of the classes, while a flat distribution indicates that the model has no information at all about the class membership of the input.

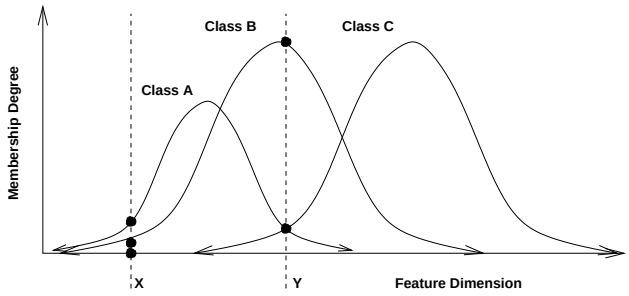

Fig. 1. Information as carried by membership functions: point X has very similar membership degrees across each class, implying that the model contains very little information about this input. The model is much more "informative" about the association of point Y with class B.

Measuring the degree of unevenness of a distribution $\Lambda = \{\lambda_c, c \in \mathcal{C}\}$, $(\sum \lambda_i = 1)$ has been tackled from different perspectives [21,40]. In particular, information theory introduced the concept of entropy [41], subsequently applied to rule-based systems [9]:

Entropy

$$I(\Lambda) = -\sum_c \lambda_c \log \lambda_c$$

The entropy is symmetric in its arguments, ensuring that no class is given more importance than any other. It is maximized when the distribution is even, and identically zero when any λ_i equals 1, effectively measuring the degree of randomness of the distribution.

By defining a normalized membership degree and a normalized information index as:

Normalized Membership Degree

$$\lambda_C(\boldsymbol{x}) = \frac{\mu_C(\boldsymbol{x})}{\sum_{c \in \mathcal{C}} \mu_c(\boldsymbol{x})}$$

Normalized Information Index

$$\mathcal{I}(\boldsymbol{x}) = \frac{\log|\mathcal{C}| - I(\Lambda)}{\log|\mathcal{C}|}$$

$\mathcal{I}(\boldsymbol{x}) \in [0,1]$ is a pointwise measure of the information the classifier provides about a given input $\boldsymbol{x}$.

2.2 Class-based Information Measure

Entropy can quantify the information a classifier provides about an input pattern. Similarly, information measures that relate to the output classes can be defined. A simple description of the impact of the classifier on a given class C over a domain D can be constructed using its average membership degree.

Average Membership Degree

$$V(C) = \frac{\int_D \mu_C(\boldsymbol{x})\, d\boldsymbol{x}}{\int_D d\boldsymbol{x}}$$

Note that the integration is carried over $d\boldsymbol{x}$, which is not necessarily uniform over D, if a prior on the feature distribution exists. Assuming normalized membership functions, a higher average membership degree to class C indicates a more uniformly distributed class over the input space. An output class represented by a membership function which takes value $+1$ everywhere on the input domain has average membership degree $+1$. A membership function with average value $V(C) = 0$ indicates an output class that is never related with any pattern of the input domain D.

2.3 Global Information Measure

In order to quantify the information contained in the whole classifier, all average membership degrees from the different membership functions should be considered together. Models with high informational content will have their average membership degrees spread across classes, while models with no information - i.e. all inputs map to a single class - will have their membership degrees unevenly distributed across classes. As previously, entropy metrics can be used to quantify this degree of unevenness. Define the relative average membership degree as follows:

Relative Average Membership Degree

$$v(C) = \frac{V(C)}{\sum_{c \in \mathcal{C}} V(c)}$$

The information contained in the classifier over the domain D for the set of classes $\mathcal{C}$ can be computed as [3, chapter8] [5]:

Global Model Information

$$I(\mathcal{C}) = -\sum_{c \in \mathcal{C}} v(c) \log v(c)$$

2.4 Distance Between Models

In order to compare alternative linguistic descriptions of the same data, it is useful to define a metric on the model space. Consider a model m_1 described by the relative average membership degree $v_1(c), c \in \mathcal{C}$ over the input domain D_1, and a second model m_2 described by $v_2(c), c \in \mathcal{C}$ over the input domain D_2. Define the divergence between models – or relative information – as:

Divergence between Models

$$\mathcal{D}(m_1 \| m_2) = \sum_{c \in \mathcal{C}} v_1(c) \log \frac{v_1(c)}{v_2(c)}$$

This divergence doesn't exactly define a metric on the model space, because it lacks the symmetry in its arguments:

$$\mathcal{D}(m_1 \| m_2) \neq \mathcal{D}(m_2 \| m_1)$$

However, it bears the other properties expected from a distance measure:

- Non-negativity: $D(m_1 \| m_2) \geq 0$
- Null kernel: $D(m_1 \| m_2) = 0 \iff \forall c \in \mathcal{C}, v_1(c) = v_2(c)$

The importance of this measure lies in the fact that there is no need for two models to share the same input space to be comparable. For example, define a hypothetical model U describing $\mathcal{C}$, which assigns equal relative average membership degrees $u(c) = 1/|\mathcal{C}|$ to each class. According to the definition of the global model information, this model has very high information, and indeed one can show that it is maximal over all models describing $\mathcal{C}$:

$$I_U(\mathcal{C}) = log|\mathcal{C}|$$

The global information of a given model m can now be reinterpreted in terms of its distance to this idealized model U:

$$I_m(\mathcal{C}) = I_U(\mathcal{C}) - \mathcal{D}(m \| U)$$

The topology induced by this divergence over the model space brings new dimensions to the analysis. Instead of simply evaluating models in terms of how they perform, they can now be compared in terms of the information they bring. Two otherwise good models can be redundant, and will not improve performance when combined. A model that is poor in isolation can be combined with models it is "distant" with, and improve significantly the performance.

2.5 Information Measure on Features

A measure of the information contained in the features can be derived from the global model information. For a given model m, define m_x as the model obtained by intersecting D with the hyperplane $\{f = x\}$. The conditional information contained in m at point x can be defined as:

$$I_m(\mathcal{C}|f = x) = I_{m_x}(\mathcal{C})$$

The information still available in the model after feature f has been exploited is thus:

Conditional Information

$$I(\mathcal{C}|f) = \int_x I(\mathcal{C}|f = x)\, dx$$

The difference between the information contained in the model before and after exploiting the feature f is defined as the mutual information:

Mutual Information

$$\mathcal{M}(\mathcal{C}, f) = I(\mathcal{C}) - I(\mathcal{C}|f)$$

The mutual information is an indicator of how much information was introduced in the model by using this feature. The less effective the input feature f is in the original model, the closer the remaining information $I(C|f)$ is to the original information $I(C)$, resulting in a lower mutual information. The input features producing the highest mutual informations are the most effective at reducing the total information contained in the model, and are deemed more informative for the analysis.

In order to compare distinct models, the relative mutual information - or information gain - is defined as:

Relative Mutual Information

$$g(\mathcal{C}, f) = \frac{\mathcal{M}(\mathcal{C}, f)}{I(\mathcal{C})} \in [0, 1]$$

The mutual information between a model m and a feature f can also be interpreted in terms of a divergence. Consider a model m_0 constructed by

removing f from the feature set. The mutual information can be expressed as:

$$\mathcal{M}(\mathcal{C}, f) = \mathcal{D}(m \| m_0)$$

This means that evaluating the contribution of a feature to a model amounts to measuring the divergence between the model when exploiting and not exploiting the feature.

2.6 Linguistic Classes

Linguistic classes define subsets of a feature space with identical informational content.

When such classes can be defined, conditioning the information on $f = x$ for any x in linguistic class L is equivalent to conditioning the information on the class itself:

$$I(\mathcal{C}|f = x) \equiv I(\mathcal{C}|f \in L)$$

In general, computing the conditional information measure $I(C|f)$ for any feature f is not trivial. However, when there is a finite set $\mathcal{L}$ of linguistic classes for feature f, the computation can be simplified:

$$\begin{aligned} I(\mathcal{C}|f) &= \int_x I(\mathcal{C}|f = x)\, dx \\ &= \sum_{L \in \mathcal{L}} I(\mathcal{C}|f \in L) \int_{x \in L} dx \\ &= \sum_{L \in \mathcal{L}} I(\mathcal{C}|f \in L)\, \gamma_L \end{aligned}$$

The term $\gamma_L = \int_{x \in L} dx$ is a prior on the belonging of feature x to linguistic class L. Because outliers in the training data tend to induce their own linguistic classes, it is important to consider this weighting, even when there is no prior on the data in the probabilistic sense. For the purpose of down-weighting outliers, the relative total membership in linguistic class L can be used as a heuristic:

$$\gamma_L = \frac{\int_{D \cap \{f \in L\}} \mu_C(\boldsymbol{x})\, d\boldsymbol{x}}{\int_D \mu_C(\boldsymbol{x})\, d\boldsymbol{x}}$$

3 Application to Fuzzy Rule Based Systems

Fuzzy rule-based systems represent linguistic associations using a superposition of unimodal membership functions that define a collection of fuzzy sets over the feature space. As a consequence, mathematical treatment of these rules translates into simple geometric manipulations.

3.1 Geometric Interpretation

Computation of the various information measures is much simplified when using a superposition of fuzzy rules as membership functions. The average membership degree to the union and the intersection of fuzzy sets derives from the min/max-definitions of intersection and union of fuzzy sets [1]:

$$V\left(\bigcup_{i=1}^{K} C^i\right) = \sum_{i=1}^{K}\left[V(C^i) - \sum_{j=i+1}^{K} V\left(C^i \cap C^j\right)\right]$$

If trapezoids are adopted as membership functions, the average membership degree to each fuzzy subset C^i can be computed from the trapezoid height h and the coordinate vectors of its vertices in the n-dimensional input space $< \boldsymbol{a}^i, \boldsymbol{b}^i, \boldsymbol{c}^i, \boldsymbol{d}^i >$:

$$V(C^i) = \frac{h\left[\prod_{j=1}^{n}\left(d_j^i - a_j^i\right) + \prod_{j=1}^{n}\left(c_j^i - b_j^i\right)\right]}{2\int_D d\boldsymbol{x}}$$

3.2 Linguistic Classes

Each intersection between trapezoids of different classes define a classification boundary. The boundary is located at:

- the intersection of their sides, if trapezoids overlap only on the sides,
- the middle point of their cores, if they overlap in their core areas,
- the middle point between the trapezoids, if they do not overlap anywhere.

Owing to the geometry of the trapezoids, each of these boundaries is a hyperplane orthogonal to one of the features in the feature set. The complete set of intersections in the model defines a partition of the input space into a finite collection of regions, each mapping a portion of the input space to a given class (Figure 2).

The projection of these decision boundaries along input feature f leads to the definition of a finite collection of thresholds that separate contiguous classes along this dimension. This set of thresholds summarizes all the information the feature is providing to the classifier: in between two of these thresholds, the membership association of the input is uniquely determined, regardless of the actual value of the input along f. As a consequence, these regions define linguistic classes as defined in Section 2.6.

The information content of each class can be computed directly by intersecting the domain D with the slice $\{f \in L\}$, leading to an efficient algorithm to compute the mutual information between the model and the input feature.

The constraint that the model only allows a finite set of decision boundaries orthogonal to a given feature is usually considered a limitation of such

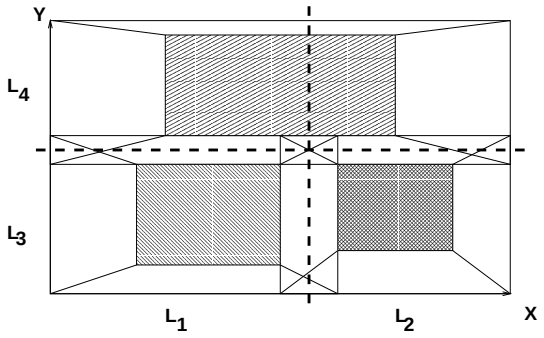

Fig. 2. Example of 3 classes defining a partition of the input space into regions separated by decision thresholds. Projected on the feature vectors, these decision thresholds define a set of linguistic classes L_1 to L_4.

models, especially in high-dimensional spaces where natural clusters tend to be sphere-shaped. Here, this property is the key to make the conditional information computations tractable.

The algorithm [5,42] is as follows:

- `Find all the` k `possible decision hyperplanes orthogonal to` f `by inspection of the possible rule combinations`[1].
- `Sort the hyperplanes by increasing order on` f.
- `For each of the` $k+1$ `linguistic classes` L_i
  - `Build the class model by intersecting the rule set with` $D \cap \{f \in L_i\}$.
  - `Compute the linguistic class information` $I(\mathcal{C}|f \in L_i)$.
- `Average all the conditional informations into` $I(\mathcal{C}|f)$.
- `Compute the mutual information` $\mathcal{M}(\mathcal{C}, f)$.

4 Application to Feature Selection: Classification of Phonetic Properties

4.1 Phonetic Properties

Phonetic properties have been used successfully in ASR [43,44], as well as in speech synthesis [45–47], as intermediate targets for general speech modeling and classification.

Each phone can be seen as the realization of a number of phonetic properties. A common example of such property is the distinction between *vowels*, *approximants* and *consonants*. Vowels in turn can be classified as *high*, *mid* or *low*, depending on the position of their first formant on the spectrum. On

[1] Note that partial ordering of the rules in the feature space as a preprocessing step can help discard irrelevant rule combinations.

another dimension, they can be regarded as *front*, *central* or *back*, depending on the place where the vowel articulation occurs. As many as 28 different properties can be defined for American English phonemes.

Table 1. Example of characterization of phonetic units

	anterior	consonantal	liquid	sonorant	voiced	stop
/a/	n/a	-	-	+	+	n/a
/l/	+	+	+	+	+	-
/p/	+	+	-	-	-	+

In order to describe a complete spoken utterance in terms of phonetic properties, a *silence* class is also often introduced, broadening the definition of a phone. A complete characterization of phonetic units from an acoustic and articulatory perspective can be found in [48,49,37,38].

4.2 Speech Frame Classification

The goal of the experiments performed in this section is to binary classify speech frames as belonging to each phonetic property class or not.

The training data consists of 20000 American English utterances, collected over the telephone and sampled at 8 kHz. An additional 8000 utterances from distinct speakers in distinct environments are used for testing. All data is balanced by gender, microphone type and noise conditions. A speech recognizer is used to align the utterances with the corresponding phones using Viterbi alignments [50] on hand-labeled transcriptions (Figure 3). The phones are themselves binary labeled depending on whether they belong or not to each of the 29 phonetic classes that were defined[2].

The signal is segmented into 10 ms frames. Each phonetic class is assigned 1000 frames for training, and 1000 frames for testing. 12 Mel Filter Cepstral Coefficients (MFCC) [51] are extracted from the signal for each frame. A 39-dimensional input feature set - typical of ASR systems, is created from the MFCC (Table 2). A fuzzy classifier [52] is trained on this feature set for each of the binary classification tasks.

4.3 Feature Selection Based on Mutual Information

After the fuzzy models are trained on the available data and for each classification task, the resulting fuzzy models are analyzed in terms of relative mutual information, as described in Section 2.5 and Section 3.

[2] Class definition by Corey Miller, synthesizing data from [37], [45] and [48]

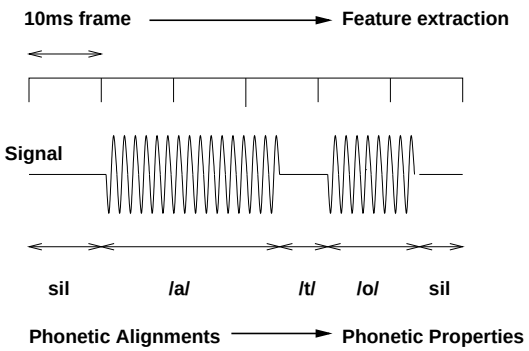

Fig. 3. Diagram of the signal processing involved: The signal is labeled phonetically in order to derive the phonetic properties, and feature extraction is performed on the signal frames to generate the inputs.

Table 2. MFCC feature vector structure

12	Cepstral Coefficients	c_1 to c_{12}
12	First Order Derivatives	∂c_1 to ∂c_{12}
12	Second Order Derivatives	$\partial^2 c_1$ to $\partial^2 c_{12}$
1	Frame Energy Coefficient	ϵ
1	First Order Energy Derivative	$\partial \epsilon$
1	Second Order Energy Derivative	$\partial^2 \epsilon$

The 39 input features are ranked according to their relative mutual information. The input features with the lowest figure of merit are also the least relevant for the fuzzy model. Although the information contained in a feature is computed with respect to a given model, it is expected to be a stable measure of the feature discriminative quality for any model taking advantage of it.

The most natural way to take advantage of the mutual information for feature selection is to decide on a threshold below which the features are to be discarded. This selection can be performed in several ways.

The most obvious criterion is to choose an absolute threshold in $[0, 1]$. The value of the threshold is decided a priori. However, not all problems have the same distribution of information across the input features, and not all sets of fuzzy rules take advantage of the same input features. An absolute threshold might penalize those systems in which the information is distributed across many input features.

Another method involves defining the threshold as a proportion of the maximum normalized mutual information value. In this way, only the features with very low information with respect to the most informative ones will be discarded.

Finally, a last strategy consists of defining the threshold based on a percentile in the mutual information histogram. This method relies on the complete distribution of the mutual information across features to base its decision, and can be expected to be more robust across variations in the mutual information spread.

After removing the input features with normalized mutual information value below the threshold, the fuzzy classifier is retrained with the remaining input features to optimize its performance.

4.4 Speech vs. Silence

The first analyzed classification task consists of speech vs. silence detection.

Speech vs. silence discrimination is a much harder signal processing problem than one could suspect initially. The amount of variability characterizing silence is much greater than for any other acoustic class. At the same time oral stops such as /p/ and /b/ are mostly constituted of silence, and are characterized mostly through their effects on neighboring phonetic units. Fricatives such as /s/ and /ch/ have a broad spectrum very confusable with white noise, especially in low bandwidth signal.

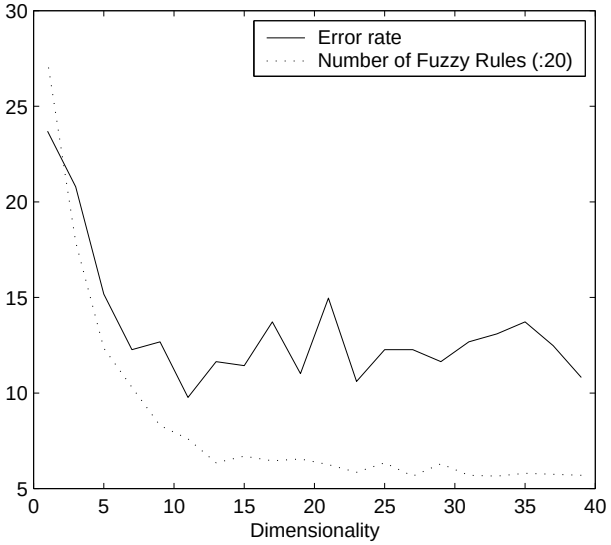

Fig. 4. Effects of input dimensionality reduction on number of rules and accuracy: speech vs. silence classification. The number of parameters (# rules × # dimensions) is dramatically reduced while not degrading the classifier accuracy

A great proportion of a speech signal is made of silence, which makes the correct recognition of silence very important for the good performance of an

ASR system. For this reason the classification of speech and silence is a very critical preliminary to any form of phonetic classification.

The classification error rate of the trained fuzzy model (Figure 4) is stable up until 10 input features are left. As expected, the energy ϵ is the predominant feature in this task ($g(\epsilon) = 0.63$). Because long silence segments are quite stationary, and because voiceless low-energy segments in the speech signal tend to be intertwined with louder segments, the first energy derivative $\partial\epsilon$ is also a very good indicator of the presence of speech ($g(\partial\epsilon) = 0.41$). Also in the highest scoring features are the lower cepstra c_1, c_2, c_3, c_5, the lower second derivatives $\partial^2 c_1$, $\partial^2 c_2$, $\partial^2 c_3$, and one first derivative ∂c_8. Conversely, the worst performing features are consistently related to the higher cepstra (c_6, c_8, $c_1 2$, ∂c_{11} and $\partial^2 c_{10}$).

4.5 Dental vs. Alveolar

Dental consonants are characterized by a constriction made against the upper teeth (examples: /dh/ as in "the", /f/), while for alveolars the constriction is made against the alveolar ridge (example: /d/). Distinguishing between the two classes can be quite difficult. Some strongly accented speakers sometimes pronounce some alveolars as dentals.

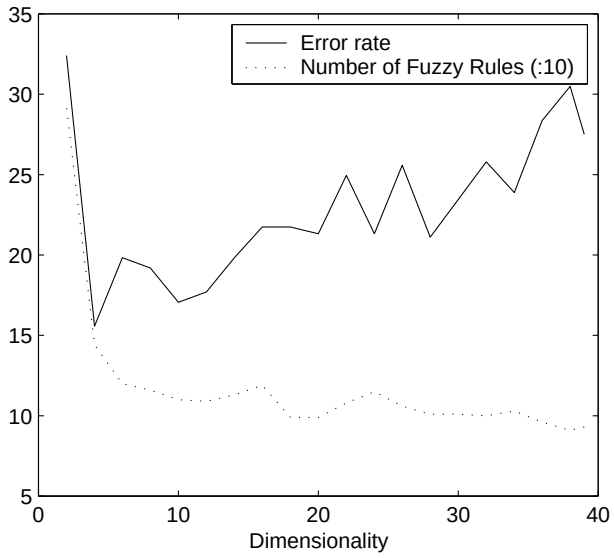

Fig. 5. Effects of input dimensionality reduction on number of rules and accuracy: dental vs. alveolar. The dimensionality reduction is associated with a dramatic improvement in accuracy

As shown in Figure 5, the reduction in dimensionality leads to a dramatic improvement in classification accuracy, due to the elimination of noisy and otherwise uninformative features. The features selected are high cepstra c_9, c_{11}, and the energy derivatives $\partial\epsilon$ and $\partial^2\epsilon$.

4.6 Oral Stops vs. Consonants

The classification of oral stops against all other consonants is a demonstration of the limitations of relying solely on the model to perform feature relevance analysis. In this situation, the amount of variability within each class (9 distinct phones are stops, while 14 are not) is too much for the classifier to handle given the amount of training data (Figure 6). The baseline error rate is about 40% using all 39 input features, showing that the system was unable to accurately learn the classification.

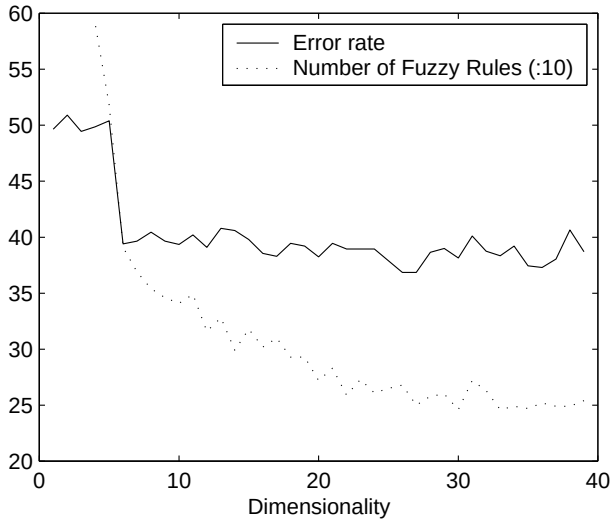

Fig. 6. Effects of dimensionality reduction on number of rules and accuracy: oral stops. The baseline model is extremely poor (about 40% error rate), and the analysis fails to recognize that one feature (the energy) is critical to the classification, and ranks it in 5^{th} position.

The analysis ranks the following features in the top 6 in decreasing order of importance: $g(c_1) = 0.53$, $g(\partial c_3) = 0.51$, $g(\partial^2 c_9) = 0.50$, $g(\partial^2 c_4) = 0.44$, $g(\partial^2\epsilon) = 0.42$, $g(\epsilon) = 0.41$. As the error profile shows, this is inaccurate. With energy ϵ as a feature, the classifier performed at an error rate of about 40%. However, without ϵ, the error rate goes suddenly to 50%, which is as good as chance. This means that all the better scoring features were irrelevant in the

absence of ϵ. Indeed, a model trained using the energy as sole feature still gives an error rate of about 42%.

This experiment stresses the fact that relying on the model implies that the model needs to be a good summary of the linguistic content in the data. If the model is not adequate in the first place, the analysis might not be able to distinguish limitations of the model from properties of the data.

Note that the total number of multidimensional rules in the model appears, in all the examples above, to be a reliable indicator of the performance to expect from a model. While the number of rules barely grow as non-meaningful dimensions are pruned out, it starts growing at a fast rate when meaningful features begin to be eliminated, as evidenced by the increase in error rate. This feature can be used as a trigger to determine when a pruned model's performance is to be questioned.

4.7 Threshold Selection

In feature selection, the choice of a threshold determines the tradeoff between input dimensionality reduction and error rate. The problem is to be able to determine the appropriate threshold directly from the mutual information of the different features, without having to retrain models.

Several selection strategies can be considered:

1. set a hard threshold on the mutual information,
2. set a threshold relative to the highest mutual information in the model,
3. set a threshold based on percentile in the mutual information histogram.

When the error rates at all the possible thresholds are known, it is possible to pick the most adequate one by manual inspection of the tradeoff curves, and label the features as informative or not.

Figure 7 compares the three automated strategies in terms of precision and recall against selection of the threshold by visual inspection:

- *recall* measures the percentage of the informative features that were selected by a given strategy,
- *precision* measures what percentage of the selected features were actually informative, according to the manually selected threshold.

Were the selection to be optimal, both these statistics would be at 100%.

Strategy 3 outperforms slightly strategies 1 and 2. Setting a hard threshold on the mutual information is brittle, since it doesn't prevent the system from having no feature selected at all. Simply normalizing the mutual information spread fixes only that issue, without improving the selectivity. On the other hand, taking advantage of the complete distribution of the mutual information across features is robust to any kind of mutual information spread and provides a practical metric for an automatic tuning of the threshold.

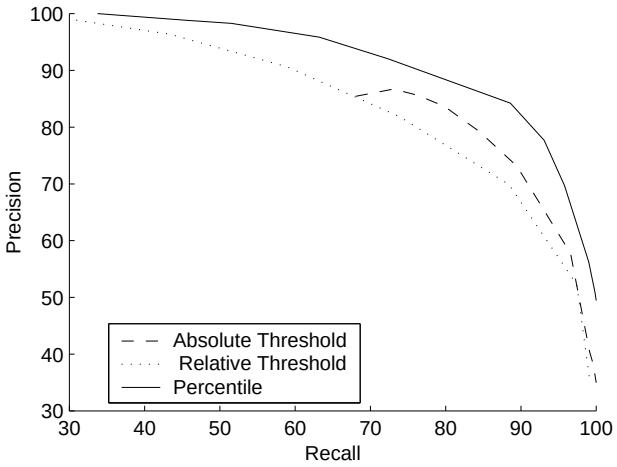

Fig. 7. Precision and recall of 3 automated threshold selection strategies, averaged across the 29 classification tasks. The optimal tradeoff lies at the upper right corner of the graph. The best method uses a feature selection threshold based on the histogram of the mutual information across all features.

4.8 Results

Figure 8 shows the average accuracy of the classifier when a global percentile threshold is used across all classification tasks.

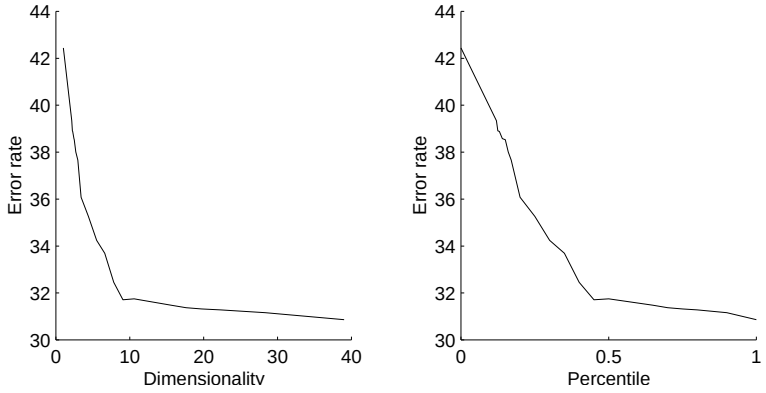

Fig. 8. Error rate vs. dimensionality across the 29 classification tasks (left). The dimensionality reduction corresponds to a threshold selection based on a percentile of the mutual information across all features (right).

The overall dimensionality can be divided by 4 on average across the 29 tasks with less than 1% loss in accuracy, by selecting the 50% percentile in the

mutual information histogram. The average number of scalar parameters in the models goes from 3274 down to 1031, i.e. a 68% reduction in complexity.

Globally, no feature is consistently deemed irrelevant across all classification tasks. The average normalized mutual information for all features is 0.2 ± 0.05, which indicates that on average all features are relevant to the task. This conclusion is supported by the very widespread use made of these features across the ASR community. In noise-free environments for example, information contained in the first and second order derivatives is of great importance, because it compensates for a lack of temporal structure inherent to the typical Hidden Markov Models [53] that are generally used. On the other hand, derivatives, especially of the higher cepstra, are also the most subject to distortion by noise, downgrading their relative importance in very noisy environments.

5 Conclusion

Information theory provides an extensive set of tools aimed at the meta-analysis of models. These tools operate on the model parameters rather than on the data directly. This leads to efficient and robust algorithms describing the information flow from features to output classes. In many situations, such indirect methods are preferable to the direct analysis of the data, especially when it comes to evaluating the relevance of a large number of parameters governing a classification model.

In this chapter, a figure of merit for feature relevance, based on a fuzzy classification of the input space, was presented. Due to the low computational load of fuzzy systems, the information contained in a set of fuzzy rules can easily be quantified. This figure of merit is based on an estimation of the mutual information, measured as a difference between the information contained in the fuzzy model before and after a given input feature is used for classification.

A feature selection algorithm is implemented, which ranks input features according to their mutual information, and discards all features deemed irrelevant by a threshold criterion. Several strategies are investigated to define the optimal threshold for this feature selection process. The best criterion is based on discarding all features above a given percentile in the mutual information histogram across inputs.

The feature selection algorithm is shown to perform well on challenging real data in high dimensions. In particular, it has been used to evaluate the contribution of commonly used speech input features to the classification of speech segments into phonetic properties, achieving an average fourfold reduction in the dimensionality with minimal accuracy cost. Future work includes benchmarking its performance against alternative feature selection techniques on similar data.

6 Acknowledgments

The authors wish to thank Michael Berthold for giving us access to his fuzzy trainer, as well as Corey Miller, Chai-Shune Hsu and Francoise Beaufays for providing invaluable data and advice.

References

1. L.A. Zadeh. Fuzzy logic and approximate reasoning. *Synthese*, Vol. 30, pp. 407-428, 1975
2. L.A. Zadeh, A fuzzy-algorithmic approach to the definition of complex or imprecise concepts, *International Journal on Man-Machine Studies*, Vol 8, pp. 249-291, 1976.
3. M. Berthold, D. Hand (Eds), *Intelligent Data Analysis, An Introduction*, Springer-Verlag, 1999
4. H. Liu and H. Motoda. *Feature Extraction, Construction and Selection: A Data Mining Perspective*, Kluwer Academic Publishers, 1998
5. R. Silipo and M. R. Berthold. Input features' impact on fuzzy decision processes. In *IEEE Transactions on Systems, Man, and Cybernetics*, part B, Vol. 30, #6, p. 821, December 2000
6. J.R. Quinlan, Induction of decision trees, in *Machine Learning*, pp. 81-106, 1986
7. J.R. Quinlan, *C4.5: Programs for Machine Learning*, Morgan Kaufmann Publishers, 1993
8. R. Silipo and G. Di Fatta, Learning to reason about data: the spring school on intelligent data analysis, *Intelligent Data Analysis journal*, Vol. 5, #5, to appear, 2001
9. C.Z. Janikow, Fuzzy decision trees: issues and methods, *IEEE Transactions on Systems, Man, and Cybernetics*, part B, Vol 28, pp. 1-14, 1998
10. Center for Spoken Language Understanding, Dept. of Computer Science Engineering, Oregon Graduate Institute, Corvallis, "Stories Corpus", Release 1.0, 1995
11. PhysioBank: `http://www.physionet.org/resources.html`
12. M.R. Jarvis and P.P. Mitra. Sleep apnea classification based on frequency of heart-rate variability, In *Proceedings of Computers in Cardiology*, 2000
13. National Center for Biotechnology Information: `http://www.ncbi.nlm.nih.gov`
14. M. McClelland et al. Complete genome sequence of Salmonella enterica serovar Typhimurium LT2, Nature 413 (6858), pp. 852-856, 2001
15. GenBank: `http://www.ncbi.nlm.nih.gov/Database/`
16. A. Inselberg and B. Dimsdale. Multidimensional iines I: representation, *SIAM J. Applied Math*, Vol. 54 (2), pp. 559-577, 1994
17. A. Inselberg and B. Dimsdale. Multidimensional lines II: representation, *SIAM J. Applied Math*, Vol. 54 (2), pp. 578-596, 1994
18. M. R. Berthold and L. O. Hall. Visualizing fuzzy points in parallel coordinates, *Technical Report UCB/CSD-99-1082*, University of California at Berkeley, 1999

19. J. Li, R.M. Gray and R.A. Olshen. Joint image compression and classification with vector quantization and a two dimensional hidden Markov model. In *Proceedings of the 1999 IEEE Data Compression Conference (DCC)*, pp. 23-32, Snowbird, Utah, March 1999
20. N. Chaddha, K. Perlmutter and R.M. Gray. Joint image classification and compression using hierarchical table-lookup vector quantization. In *Proceedings of the 1996 IEEE Data Compression Conference (DCC)*, J.A. Storer and M. Cohn, editors, IEEE Computer Society Press, Snowbird, Utah, March 1996
21. C. Apte, S.J. Hong, J.R.M. Hosking, J. Lepre, E. Pednault, and B.K. Rosen. Decomposition of heterogeneous classification problems. *Intelligent Data Analysis Journal*, Vol 2, #2, 1998
22. A. Webb. *Statistical Pattern Recognition*, Chapter 8: Feature selection and extraction, Arnold, London, 1999
23. S. Salzberg. Improving classification methods via feature selection. *John Hopkins Technical Report*, 1992
24. M. Dash and H. Liu. Feature selection for classification. *Intelligent Data Analysis*, 1, (3), 1997
25. Y. Yang and J.O. Pedersen. A comparative study on feature selection in text categorization. In *Procccedings of the 14th International Conference on Machine Learning*, pp. 412-420, Morgan Kaufmann, 1997.
26. M.A. Hall. Correlation-based feature subset selection for machine learning. PhD dissertation, Department of Computer Science, University of Waikato, 1999
27. B. Pfahringer. Compression-based feature subset selection. In *Proceedings of the IJCAI-95 Workshop on Data Engineering for Inductive Learning*, pp. 109-119, Montreal, Canada, 1995
28. P. Turney, Canada National Research Council Online Bibliography `http://extractor.iit.nrc.ca/bibliographies/feature-selection.html`
29. K. Koumpis and S. Renals. The role of prosody in a voicemail summarization system. In *Proceedings of the ISCA Workshop on Prosody in Speech Recognition and Understanding*, pp. 87-92, 2001
30. G. John, R. Kohavi and K. Pfleger. Irrelevant features and the subset selection problem. In *Machine Learning: Proceedings of the Eleventh International Conference (ICML-94)*, pp. 121-129, New Brunswick, NJ, Morgan Kaufmann, 1994
31. D.W. Aha and R.L. Bankert. A comparative evaluation of sequential feature selection algorithms. In *Artificial Intelligence and Statistics V*, D. Fisher and J.-H. Lenz, editors, Springer-Verlag, New York, NY, 1996
32. D.W. Aha and R.L. Bankert. Feature selection for case-based classification of cloud types: An empirical comparison. In *Proceedings of the 1994 AAAI Workshop on Case-Based Reasoning*, pp. 106-112, AAAI Press, Seattle, WA, 1994
33. H. Almuallim and T.G. Dietterich. Learning with many irrelevant features. In *Proceedings of the Ninth National Conference on Artificial Intelligence*, pp. 547-552, Menlo Park, CA, AAAI Press, 1991
34. K. Kira and L.A. Rendell. A practical approach to feature selection. In *Proceedings of the Ninth International Workshop on Machine Learning*, pp. 249-256, Aberdeen, Scotland, Morgan Kaufmann, 1992

35. J. Casillas, O. Cordón, M. J. del Jesus, F. Herrera. Genetic Feature Selection in a Fuzzy Rule-Based Classification System Learning Process for High Dimensional Problems. *Information Science*, #136, pp. 169-191, 2001
36. F. Jelinek. *Statistical Methods for Speech Recognition*, MIT Press, Cambridge, 1998
37. P. Ladefoged. *A course in phonetics*, 1975
38. P. Ladefoged and I. Maddieson. *The sounds of the world's languages*, 1996
39. S. Chang, S. Greenberg and M. Wester. An elitist approach to articulatory-acoustic feature classification. In *Proceedings of Eurospeech 2001*, 2001
40. I. Kononenko. On biases in estimating multivalued attributes. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, pp. 1034-1040, 1995
41. T.M. Cover and J.A. Thomas. *Elements of Information Theory*, John Wiley & Sons, New York, NY, USA, 1991
42. R. Silipo and M.R. Berthold. Discriminative power of input features in a fuzzy model. In *Proceedings of the Third International Symposium on Intelligent Data Analysis (IDA-99)*, D. Hand, J. Kok, M. Berthold, editors, *Advances in Intelligent Data Analysis" (IDA-99), Lecture Notes in Computer Science*, LNCS 1642, pp. 87-98, Springer-Verlag, 1999
43. S. King, T. Stephenson, S Isard, P. Taylor and A. Strachan. Speech recognition via phonetically featured syllables. In *Proceedings of the 5th International Conference on Spoken Language Processing (ICSLP'98)*, 1998
44. M. Richardson, J. Bilmes and C. Diorio. Hidden-articulator markov models for speech recognition In *Proceedings of ISCA ASR2000*, pp. 133-137, Paris, France, 2000
45. C.A. Miller, *Pronunciation Modeling in Speech Synthesis*, University of Pennsylvania, 1998
46. T. Sejnowski and C.R. Rosenberg. NETtalk: a parallel network that learns to read aloud. *Johns Hopkins University Technical Report*, JHU/EECS-86/01, 1986
47. T. Sejnowski and C.R. Rosenberg. Parallel networks that learn to pronounce English text. *Complex Systems*, #1, pp. 145-168, 1987
48. N. Chomsky and M. Halle. *The Sound Pattern of English*, MIT Press, 1968
49. L.M. Hyman. *Phonology: Theory and Analysis*, 1975
50. G. D. Forney. The Viterbi Algorithm. In *Proceedings of the IEEE*, pp. 268-278, 1973
51. S. B. Davis and P. Mermelstein. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. ASSP-28, #4, p. 357, 1980
52. K.P. Huber, M.R. Berthold. Building precise classifiers with automatic rule extraction. In *IEEE International Conference on Neural Networks*, Vol. 3, pp. 177-184, 1995
53. X. D. Huang, Y. Ariki and M. A. Jack. *Hidden Markov Models for Speech Recognition*, Edinburgh University Press, Edinburgh, 1990.