Pattern Recognition Letters 66 (2015) 71–80

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

Maximum-likelihood normalization of features increases the robustness of neural-based spoken human-computer interaction ☆

Edmondo Trentin*

DIISM - University of Siena, V. Roma 56, Siena I-53100, Italy

ARTICLE INFO

Article history:
Available online 15 July 2015

Keywords:
Feature normalization
Maximum likelihood estimation
Neural network
Automatic speech recognition
Hidden Markov model

ABSTRACT

Robust acoustic modeling is essential in the development of automatic speech recognition systems applied to spoken human-computer interaction. To this end, traditional hidden Markov models (HMM) may be improved by hybridizing them with artificial neural networks (ANN). Crucially, ANNs require input values that do not compromize their numerical stability. In spite of the relevance feature normalization has on the success of ANNs in real-world applications, the issue is mostly overlooked on the false premize that "any normalization technique will do". The paper proposes a gradient-ascent, maximum-likelihood algorithm for feature normalization. Relying on mixtures of logistic densities, it ensures ANN-friendly values that are distributed over the (0, 1) interval in a uniform manner. Some nice properties of the approach are discussed. The algorithm is applied to the normalization of acoustic features for a hybrid ANN/HMM speech recognizer. Experiments on real-world continuous speech recognition tasks are presented. The hybrid system turns out to be positively affected by the proposed technique.

1. Introduction

The development of systems for spoken human-computer interaction [33] per se revolves around two major components, namely a speech synthesizer and an automatic speech recognizer. The latter, in turn, involves a language model and a so-called acoustic model. While the language model aims at constraining the hypothesis space within grammatically or probabilistically plausible borders (e.g., by means of *n*-grams estimated via frequentist approaches from text or transcripts of dialogues [7]), the acoustic model copes with the speech signal, aiming at capturing the significant statistical properties of the acoustic phenomena at a phonetic level of analysis [32]. This is not accomplished on the raw waveform, but on a suitable representation of the signal in the form of a set of acoustic parameters, or features. These features are extracted via pre-defined signal processing techniques, mostly relying on spectral analysis and filtering [12].

Hidden Markov models (HMM) [17,28] are far the most popular model at the acoustic level. Although they are viable approaches to the problem of acoustic modeling (allowing for good recognition performance under many circumstances) they also suffer from some severe limitations, mostly due to their parametric nature. Drawbacks of HMMs are analyzed in detail, for instance, in [5,35]. These limitations are the reason why several scientists have been proposing hybrid systems that combine the long-term modeling capabilities of HMMs and the flexibility [10] of artificial neural networks (ANN) [3]. Albeit traditional ANNs (e.g., multilayer perceptrons and radial basis function networks [3]) resulted infeasible for speech recognition when used as stand-alone recognizers, they proved themselves successful in several combinations with HMMs. Established instances are surveyed in [35].

A ticklish aspect in using ANNs within hybrid paradigms (as well as in using them in general) lies in the numeric nature, distribution, and range of the input features. While continuous-density HMMs, based on mixtures of Gaussian components, are little (or, not at all) sensitive to these issues, ANNs require input values that do not compromize their numerical stability. Indeed, normalization of the feature vectors is common (and, necessary) practice in real-world applications of ANNs. Its relevance and its potential implications have long been investigated in classic pattern recognition literature [13,14,18]. Although the success of connectionist models in difficult tasks may depend on the outcome of a more or less adequate feature normalization, surprisingly enough the topic is often under-rated on the false premize that "any normalization technique will just do". The paper copes with the issue, proposing a novel, ANN-oriented feature normalization algorithm that is then applied to a continuous speech recognition task relying on the ANN/HMM hybrid paradigm we proposed in [36]. This paper stems from some preliminary ideas we put forward in a conference presentation [34], and introduces a new

☆ This paper has been recommended for acceptance by Friedhelm Schwenker.

* Tel.: +39 577 233601; fax: +39 577 233602.
E-mail address: trentin@dii.unisi.it

http://dx.doi.org/10.1016/j.patrec.2015.07.003
0167-8655/

algorithm for an improved model (whose properties are discussed, as well), along with a set of new, sound experiments.

1.1. Feature normalization: formulation of the problem

Let us assume that the input (or, output) patterns are in the form $\mathbf{x} = (x_1, \ldots, x_d)$, and that they are extracted from a d-dimensional, real-valued feature space $X \in \Re^d$. Individual feature values x_i, $i = 1, \ldots, d$, are measurements of certain attributes, according to a problem-specific feature extraction process. Such measurements are expressed, in general, in terms of different units, and the latter ones may span different possible ranges of values. Major motivations for applying a normalization method include the following.

1. Reducing all features $x_1, \ldots, x_d$ to a common range (a, b), where $a, b \in \Re$. In so doing, increased homogeneity of values is gained, yielding a common (e.g., Euclidean) "distance measure" over patterns along the different axis. Furthermore, all features are given the same credit, or weight: unnormalized features that span a wider numerical range would otherwise overpower features defined over smaller intervals.
2. Tackling, or reducing, numerical stability problems of the learning algorithms in the ANN (i.e., during the computation of partial derivatives of the nonlinearities of the model). In particular, input values should not exceed a certain (a, b) interval, in order to avoid the phenomenon of "saturation" of sigmoids. As a matter of fact, saturation occurs when the activation value a (input argument) of a sigmoid $\sigma(a)$ is along the tails of $\sigma(.)$, where the partial derivative $\frac{\delta\sigma(a)}{\delta a}$ is numerically null. In case of saturation, the sigmoid is basically "stuck" and it cannot provide any further contribution to the gradient-driven learning of connection weights.
3. Stabilizing the numerical behavior of the delta-rule in the back-propagation (BP) algorithm [30]. It is known that BP prescribes a delta-rule for the connection weights either in the form $\Delta w_{ij} = \eta\delta_i\sigma_j(a_j)$ (for a generic hidden or output weight w_{ij}), or $\Delta w_{jk} = \eta\delta_j x_k$ (for weights w_{jk} in the first layer, i.e. the input layer). The neuron-specific quantities δ_i, δ_j are the core of BP, and they are strictly related to the back-propagated gradient of the criterion function. A sigmoid activation function $\sigma_j(a_j)$ is associated with the generic jth hidden neuron. Assuming the same learning rate η is used in both the above forms of delta-rule, it is seen that unnormalized features x_k having large value (say, $x_k \gg 1$) would overdrive the learning process for input weights w_{jk} (whose Δ's are directly affected by x_k) w.r.t. the other weights of the ANN (whose Δ's are rather affected by $\sigma_j(a_j) < 1$).
4. Allowing for application of a nonlinear (sigmoid) output layer even though the target outputs are defined over a wider range $\mathcal{Y} \subset \Re$, $\mathcal{Y} \not\subseteq (0, 1)$. Actually, sigmoids in the form $\frac{1}{1+e^{-a}}$ are limited to the $(0, 1)$ interval, and hyperbolic-tangent sigmoids range over the $(-1, 1)$ interval, while the task at hand might require outputs exceeding these ranges.
5. Leading to data distributions that are basically invariant to rigid displacements of the coordinates.

1.2. Concise survey of established normalization methods

Traditional normalization techniques rely prevalently on the following approaches: (i) for each feature $i = 1, \ldots, d$, find the maximum absolute value M_i (i.e., $M_i \in \mathcal{R}^+$) over the training set, and normalize each pattern $\mathbf{x}$ to obtain a new pattern $\mathbf{x}'$ defined as $\mathbf{x}' = (x_1/M_1, \ldots, x_d/M_d)$. This ensures features within the $(-1, 1)$ range. A similar technique is described in [8]; (ii) compute the sample mean m_i and the sample variance s_i for each feature $i = 1, \ldots, d$, and normalize $\mathbf{x}$ to obtain $\mathbf{x}' = (\frac{x_1-m_1}{s_1}, \ldots, \frac{x_d-m_d}{s_d})$. This ensures zero mean and unit variance along all coordinate axis of the normalized feature space [18]; and (iii) transform the components of $\mathbf{x}$ by means of a smooth nonlinearity ϕ: $\Re \to \Re$ that maps its argument onto the desired, normalized range: $\mathbf{x}' = (\phi(x_1), \ldots, \phi(x_d))$. Approaches (i) and (ii), i.e. mean subtraction and division by maximum, are sometimes combined.

Other (often similar) approaches can be found in the literature. For instance, [15] presents an algorithm based on a heterogeneity measure, while [21] proposes a combined normalization/clustering procedure. Different methods rely on linear projections, e.g. the eigenvector projection or Karhunen–Loeve method [18], where the original features are linearly transformed to a lower dimensionality space. These transformations imply a certain loss of information w.r.t. the original feature space representation of patterns.

1.3. Overview of the paper

This paper introduces a novel ANN-oriented feature normalization technique that ensures values that are distributed over the $(0, 1)$ interval in a uniform manner. Thence, the data are basically reduced to a sample-invariant distribution, regardless of their original statistics. As a consequence, at least in principle, the ANN architecture and the training parameters (e.g., the learning rate), once selected properly, may be expected to fit different d-dimensional datasets. Eventually, it is straightforward to map the normalized data onto any zero-centered interval, e.g. $(-1, 1)$, if desired. The technique is inspired by an approach suggested by Yoshua Bengio[1], who used the *rank* of discrete observations as their numeric feature value. The normalization is obtained starting from a maximum-likelihood estimation of the probabilistic distribution of input features, according to a particular parametric model, namely a mixture of logistic densities (and the corresponding cumulative distribution function). Since no closed-form solution can be found for the parameters of this model, a gradient-ascent iterative algorithm is proposed. The technique is described in detail in Section 2.

The model is shown to be general enough to cover (to any degree of precision) all cases of practical interest of data distributions (according to a properly defined class of non-paltry probability density functions). In addition to the aforementioned benefits, the technique turns out to be compliant with the very numerical nature of the ANN (it is realized via a mixture of sigmoids, that can even be encapsulated within the ANN itself). These properties of the proposed approach are discussed in Section 3.

The ANN/HMM hybrid speech recognizer is reviewed in Section 4, where its architecture and learning algorithm are summarized. An experimental evaluation on real-world, speaker-independent, continuous speech recognition tasks from the *SPK*, *Aurora*, and *SpeechDat-Car/Aurora 3* corpora (Section 5) confirms that suitable ANN/HMM hybrids may outperform standard HMMs. The experiments show also that: (1) normalization of the acoustic features is necessary in order for the ANNs to behave properly, but has a negative impact on the Gaussian-based HMM, as expected; (ii) traditional normalization methods do work, but the choice of the specific technique affects profoundly the robustness of the resulting ANN/HMM-based speech recognizer; (iii) the proposed normalization algorithm is effective. It improves significantly over the traditional approaches, turning out to be noise-tolerant and more suitable to the speech recognition tasks under consideration. Conclusive remarks are drawn in Section 6.

2. The proposed normalization method

Normalization is accomplished by transforming individual components of each input pattern into the corresponding value of the cumulative distribution function (cdf) of the inputs, estimated on a

[1] Y. Bengio, personal communication to the author.