Tandem Decoding of Children's Speech for Keyword Detection in a Child-Robot Interaction Scenario

MARTIN WÖLLMER and BJÖRN SCHULLER
Technische Universität München, Germany
and
ANTON BATLINER and STEFAN STEIDL
Friedrich-Alexander-Universität Erlangen-Nürnberg, Germany
and
DINO SEPPI
Katholieke Universiteit Leuven, Belgium

In this article, we focus on keyword detection in children's speech as it is needed in voice command systems. We use the FAU Aibo Emotion Corpus which contains emotionally colored spontaneous children's speech recorded in a child-robot interaction scenario and investigate various recent keyword spotting techniques. As the principle of bidirectional Long Short-Term Memory (BLSTM) is known to be well-suited for context-sensitive phoneme prediction, we incorporate a BLSTM network into a Tandem model for flexible coarticulation modeling in children's speech. Our experiments reveal that the Tandem model prevails over a triphone-based Hidden Markov Model approach.

Categories and Subject Descriptors: I.5.1 [**Computing Methodologies**]: Pattern Recognition—*Neural Nets*; I.2.7 [**Computing Methodologies**]: Natural Language Processing—*Speech recognition and synthesis*

General Terms: Algorithms, Performance

Additional Key Words and Phrases: Children's Speech, Dynamic Bayesian Networks, Keyword Spotting, Long Short-Term Memory

1. INTRODUCTION

Offering a natural and intuitive input modality, speech interfaces for children are already used in many applications such as reading tutors or voice command systems [Hagen et al. 2007; Steidl et al. 2010]. Thus, optimizing and evaluating automatic speech recognition (ASR) techniques for children's speech is an active area

Author's address: M. Wöllmer, Institute for Human-Machine Communication, Technische Universität München, Theresienstr. 90, 80333 München, Germany
The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement No. 211486 (SEMAINE), grant No. IST-2002-50742 (HUMAINE), and grant No. IST-2001-37599 (PF-STAR). The responsibility lies with the authors.

of research [Das et al. 1998; Narayanan and Potamianos 2002; Steidl et al. 2010]. Recognition of children's speech is known to be a challenge for state-of-the-art ASR systems since acoustic and linguistic properties strongly differ from adult speech [Giuliani and Gerosa 2003]. Typical differences in pitch, formant positions, and coarticulation led to the development of techniques like voice transformations and frequency warping [Potamianos et al. 1997; Gustafson and Sjölander 2002]. In this way, adult speech recognizers can be used to transcribe children's speech as well. Other systems are directly trained on children's speech, aiming to obtain models that fit the respective application scenario [Schuller et al. 2008].

Since full spoken language understanding without any restriction of the expected vocabulary is hardly feasible and not necessarily needed in today's child-machine interaction scenarios, keyword spotting can be applied as an alternative to continuous speech recognition [Foote 1999; Schröder et al. 2008]. The aim of keyword spotting is to detect a set of predefined keywords from continuous speech signals [Rose 1995]. At present the predominant methodology for keyword spotting is using Hidden Markov Models (HMM) [Rose and Paul 1990; Benayed et al. 2003; Ketabdar et al. 2006]. Such systems have to model both, keyword and non-keyword (*garbage*) parts of the speech signal, which is difficult, since a garbage model trained to capture arbitrary speech can in principle also model the keywords. Both, the garbage and the keyword HMMs can either be whole-word models or connected phoneme models. Whole-word models can be used whenever the keywords occur frequently in the training corpus, while vocabulary independent systems that use phoneme modeling can be trained on any database, regardless of whether the keywords are contained in the corpus or not [Mamou et al. 2007]. This makes vocabulary independent systems very flexible, since new keywords can be added to the system without having to train new models - only the pronunciation of the new keyword has to be defined.

Wöllmer et al. [2009d] proposed a technique that uses a hierarchical graphical model (GM) architecture to detect keywords in continuous speech. It is based on phoneme modeling and therefore allows changes in the keyword vocabulary after model training. A second advantage of this system is that it does not require the training of an explicit garbage model but rather uses a binary *garbage variable* together with the concept of *switching parents* [Bilmes 2003] in order to distinguish keywords from arbitrary speech.

In this contribution, we will use the Dynamic Bayesian Network (DBN, [Murphy 2002]) introduced by Wöllmer et al. [2009d] as basis for Tandem modeling of children's speech, applying a context-sensitive recurrent neural network (RNN) architecture for phoneme prediction and a graphical model for keyword detection. Tandem or hybrid architectures that combine discriminatively trained neural networks with graphical models such as HMMs were shown to be applicable for speech recognition, and their popularity has grown in recent years [Boulard and Morgan 1994; Bengio 1999; Hermansky et al. 2000; Ellis et al. 2001; Ketabdar and Bourlard 2008]. However, the limitations of recurrent neural networks still prevent such hybrid or Tandem techniques from becoming a widely used standard in ASR systems. One such limitation is the so-called *vanishing gradient problem* that causes the backpropagated error in RNNs to either blow up or exponentially decay over time

[Hochreiter et al. 2001]. This strongly limits the amount of context that RNNs can access and model. Yet, due to coarticulation effects in human speech, modeling a sufficient amount of context during speech feature generation and processing is essential. On a higher level, context in speech is usually modeled via triphones and language models, while on the feature level, most ASR systems incorporate only a very limited amount of context by using first and second order regression coefficients of low-level descriptors such as Mel-Frequency Cepstral Coefficients (MFCC) as additional features.

Only few studies try to address the topic of considering a higher amount of context on the feature level [Hermansky and Fousek 2008] on the one hand, and solving the vanishing gradient problem in RNNs on the other hand [Hochreiter and Schmidhuber 1997; Jaeger 2001; Schaefer et al. 2008]. An elegant and efficient way to enable long-range context modeling with recurrent neural networks has been proposed by Hochreiter and Schmidhuber [1997] and refined by Graves and Schmidhuber [2005]: Bidirectional Long Short-Term Memory (BLSTM) networks are able to model a self-learned amount of contextual information by using memory blocks in the hidden layer of RNNs. Even though this technique was shown to prevail over the triphone principle [Graves et al. 2005], it has been applied rarely for keyword spotting so far: Wöllmer et al. [2009c] showed that the framewise phoneme predictions of BLSTM can enhance the performance of a discriminative keyword spotter [Keshet et al. 2007]; and Fernandez et al. [2007a] introduced a keyword spotter using BLSTM for whole-word modeling.

In this article we apply BLSTM modeling in order to generate phoneme predictions that are decoded together with conventional speech features in a Dynamic Bayesian Network and use this principle for keyword detection in a child-robot interaction scenario. Our technique aims at addressing the acoustic and linguistic characteristics of children's speech that typically lead to poor recognition performance when applying adult speech recognizers to children. Since the linguistic variability and the different language strategies that children use when interacting with machines cannot be adequately modeled by language models trained on large databases of adult speech, and since the number of existing spontaneous children's speech corpora tends to be too small to train a versatilely applicable children's language model, we decided for a flexible vocabulary independent word spotting strategy that exclusively relies on acoustic evidence and does not require a language model. Our system accounts for the mismatch between children's and adult speech characteristics – such as higher pitch – by using acoustic models trained on children's speech. Hence, the recognizer does not rely on preprocessing steps such as voice or feature transformation but uses acoustic models that inherently capture the properties of children's speech. As the characteristics of coarticulation in children's speech strongly differ from coarticulation effects in adult speech [Gerosa et al. 2006], we apply BLSTM networks as an efficient and comparably novel method of context modeling. Children develop coarticulation skills with increasing age which leads to strong variations in the amount of temporal context that needs to be considered to capture coarticulation for context-sensitive speech feature generation and acoustic modeling [Repp 1986; Mayo et al. 2003]. Thus, it seems inappropriate to manually define an inflexible, fixed amount of context, as it is commonly done when stacking

multiple low-level feature frames for neural network based feature generation [Grezl and Fousek 2008]. By contrast, our approach of modeling contextual information in children's speech via BLSTM networks allows us to *learn* the proper amount of relevant context.

Our technique significantly differs from past approaches towards keyword spotting via BLSTM networks: The discriminative approach of Wöllmer et al. [2009c] does not apply Markov chains to model the temporal evolution of speech, but maps the acoustic representation of an utterance along with the target keyword into an abstract vector space, using a set of feature functions that provide confidence scores based on the output of framewise phoneme classifiers. This strategy, however, is rather suited for off-line keyword search than for an on-line application of child-robot interaction since it does not operate in real-time. The disadvantage of the method proposed by Fernandez et al. [2007a] is that it is not vocabulary independent, as it has a separate output unit for each keyword.

Motivated by our preliminary experiments on keyword spotting using a Tandem BLSTM-DBN architecture for decoding read adult speech as contained in the TIMIT corpus [Wöllmer et al. 2009b], this article shows how keyword detection in children's speech can be improved by coarticulation modeling via bidirectional Long Short-Term Memory. We compare the proposed system architecture to the DBN approach introduced by Wöllmer et al. [2009d], the Connectionist Temporal Classification (CTC) method of Fernandez et al. [2007a], as well as to a conventional HMM keyword spotter and a multi-stream BLSTM-HMM system. We focus on the task of detecting keywords in emotionally colored and spontaneous German children's speech that was recorded during child-robot interaction (the FAU Aibo Emotion Corpus [Steidl 2009]). Thereby we consider a set of 25 different keywords which either correspond to command words or are relevant for recognizing the child's emotional state.

The structure of this article is as follows: Section 2 describes the FAU Aibo Emotion Corpus, Section 3 reviews the principle of Long Short-Term Memory, Section 4 introduces Connectionist Temporal Classification as applied for keyword spotting by Fernandez et al. [2007a], Section 5 explains our Tandem BLSTM-DBN keyword spotter, and Section 6 contains experiments and results.

2. THE FAU AIBO EMOTION CORPUS

The experiments described in this paper are based on the FAU Aibo Emotion Corpus, a corpus of German spontaneous speech with recordings of children at the age of 10 to 13 years communicating with a pet robot; it is described in detail in [Steidl 2009]. The general framework for this database with children's speech is child-robot communication and the elicitation of emotion-related speaker states. The robot is Sony's (dog-like) robot Aibo. The basic idea has been to combine children's speech and naturally occurring emotional speech within a Wizard-of-Oz task. The speech is spontaneous, because the children were not told to use specific instructions but to talk to Aibo like they would talk to a friend. In this experimental design, the child is led to believe that Aibo is responding to his or her commands, but the robot is actually being remote-controlled by a human operator, using the 'Aibo Navigator' software over a wireless LAN. The wizard causes Aibo to perform

a fixed, predetermined sequence of actions, which takes no account of what the child is actually saying. For the sequence of Aibo's actions, we tried to find a good compromise between obedient and disobedient behavior: We wanted to provoke the children in order to elicit emotional behavior but of course we did not want to run the risk that they discontinue the experiment. The children believed that Aibo was reacting to their orders – albeit often not immediately. In fact, it was the other way round: Aibo was always strictly following the same screen-plot, and the children had to align their orders to its actions.

The data was collected from 51 children (21 male, 30 female) aged 10 to 13 years from two different schools (*Mont* and *Ohm*); the recordings took place in the respective class-rooms. Speech was transmitted via a wireless head set (Shure UT 14/20 TP UHF series with microphone WH20TQG) and recorded with a DAT-recorder (sampling rate 48 kHz, quantization 16 bit, down-sampled to 16 kHz). The total vocabulary size is 1.1 k. Each recording session took around 30 minutes; in total there are 27.5 hours of data. The recordings contain large amounts of silence, which are due to the reaction time of Aibo. After removing longer pauses, the total amount of speech is equal to 8.9 hours. All recordings were split into turns using a pause threshold of ≥ 1 s.

In our speaker-independent experiments, we use all speech recorded at the *Ohm* school for training (6 370 turns), apart from two randomly selected *Ohm*-sessions which are used for validation (619 turns). The sessions recorded at the *Mont* school are used for testing (6 653 turns, see also Table I).

set	**school**	**turns**	**words**	**duration**
training	Ohm	6 370	22 244	4.5 h
validation	Ohm	619	2 516	0.5 h
testing	Mont	6 653	23 641	3.9 h

Table I. Size of the training, validation, and test set: school in which the children were recorded, number of turns, number of words, and duration.

3. LONG SHORT-TERM MEMORY

Since context modeling via Long Short-Term Memory [Hochreiter and Schmidhuber 1997] networks was found to enhance keyword spotting performance in natural conversation scenarios [Wöllmer et al. 2009c], our Tandem BLSTM-DBN keyword spotter uses framewise phoneme predictions computed by a bidirectional LSTM net (see Section 5). Thus, this section outlines the basic principle of the Long Short-Term Memory RNNs.

Framewise phoneme prediction presumes a classifier that can access and model long-range context, since due to coarticulation effects in human speech, neighboring phonemes influence the cepstral characteristics of a given phoneme [Bilmes 1998; Yang et al. 2000]. Consequently, when attempting to predict phonemes frame by frame, a number of preceding (and successive) speech frames have to be taken into account in order to capture relevant speech characteristics. The *number* of speech frames which should be used to obtain enough context for reliably estimating phonemes is hard to determine – especially when processing children's speech

for which coarticulation properties are not as well researched as for adult speech. Thus, a classifier that is able to *learn* the amount of context is a promising alternative to manually defining fixed time windows. Static techniques such as Support Vector Machines [Cortes and Vapnik 1995] do not explicitly model context but rely on either capturing contextual information via statistical functionals of features [Schuller et al. 2009b] or aggregating frames using Multi-Instance Learning techniques [Schuller and Rigoll 2009]. Dynamic classifiers like Hidden Markov Models are often applied for time warping and flexible context modeling using, e. g., triphones or quinphones. Yet, HMMs have drawbacks such as the inherent assumption of conditional independence of successive observations, meaning that an observation is statistically independent of past ones, provided that the values of the hidden variables are known. Hidden Conditional Random Fields (HCRF) [Quattoni et al. 2007] are one attempt to overcome this limitation. However, also HCRF offer no possibility to model a self-learned amount of contextual information. Other classifiers such as neural networks are able to model a certain amount of context by using cyclic connections. These so-called recurrent neural networks can in principle map from the entire *history* of previous inputs to each output. Yet, the analysis of the error flow in conventional recurrent neural nets led to the finding that long-range context is inaccessible to standard RNNs since the backpropagated error either blows up or decays over time (vanishing gradient problem [Hochreiter et al. 2001]). This led to various attempts to address the problem of vanishing gradients for RNN, including non-gradient based training [Bengio et al. 1994], time-delay networks [Lang et al. 1990; Lin et al. 1996; Schaefer et al. 2008], hierarchical sequence compression [Schmidhuber 1992], and echo state networks [Jaeger 2001]. One of the most effective techniques is the Long Short-Term Memory architecture [Hochreiter and Schmidhuber 1997], which is able to store information in linear memory cells over a longer period of time. LSTM networks are able to overcome the vanishing gradient problem and can learn the optimal amount of contextual information relevant for the classification task.

An LSTM layer is composed of recurrently connected memory blocks, each of which contains one or more memory cells, along with three multiplicative 'gate' units: the input, output, and forget gates. The gates perform functions analogous to read, write, and reset operations. More specifically, the cell input is multiplied by the activation of the input gate, the cell output by that of the output gate, and the previous cell values by the forget gate (see Figure 1). The overall effect is to allow the network to store and retrieve information over long periods of time. For example, as long as the input gate remains closed, the activation of the cell will not be overwritten by new inputs and can therefore be made available to the net much later by opening the output gate.

Another problem with standard RNNs is that they have access to past but not to future context. This can be overcome by using bidirectional RNNs [Schuster and Paliwal 1997], where two separate recurrent hidden layers scan the input sequences in opposite directions. The two hidden layers are connected to the same output layer, which therefore has access to context information in both directions. The amount of context information that the network actually uses is learned during training, and does not have to be specified beforehand. Forward and backward

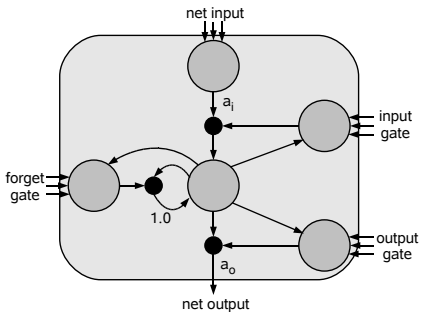


Fig. 1. LSTM memory block consisting of one memory cell: The input, output, and forget gates collect activations from inside and outside the block which control the cell through multiplicative units (depicted as small circles); input, output, and forget gate scale input, output, and internal state respectively; a_i and a_o denote activation functions; the recurrent connection of fixed weight 1.0 maintains the internal state.

context are learned independently from each other. Bidirectional networks can be applied whenever the sequence processing task is not truly on-line. For speech recognition tasks this means that unidirectional context can be used for incremental real-time decoding, while bidirectional context can be applied, e. g., at the end of an utterance to refine the recognition output once the whole utterance is available [Graves et al. 2005]. However, often a small buffer is enough in order to profit from bidirectional context, so that bidirectional networks can also be applied for causal systems whenever a short output latency is tolerable. Figure 2 shows the structure of a simple bidirectional network.

Combining bidirectional networks with LSTM gives bidirectional Long Short-Term Memory [Graves et al. 2005], which has demonstrated excellent performance in phoneme recognition [Graves and Schmidhuber 2005], keyword spotting [Fernandez et al. 2007a], handwriting recognition [Liwicki et al. 2007; Graves et al. 2008a], noise modeling [Wöllmer et al. 2009e], and emotion recognition from speech [Wöllmer et al. 2010c].

4. CONNECTIONIST TEMPORAL CLASSIFICATION

One possibility to use BLSTM networks for keyword detection is to train the network directly on the keywords, so that the network learns a mapping from speech features to keywords. Such an approach was investigated by Fernandez et al. [2007a] and will be evaluated in Section 6, where we compare the keyword spotting accuracy of this discriminative technique to our (vocabulary independent) *phoneme-based* Tandem BLSTM-DBN method.

The BLSTM networks applied by Fernandez et al. [2007a] are trained with the Connectionist Temporal Classification (CTC) objective function. CTC allows recurrent neural networks to map unsegmented sequential data onto a sequence of labels [Graves et al. 2006]. The output of a network trained with CTC typically consists of a series of spikes corresponding to keyword events that are detected in

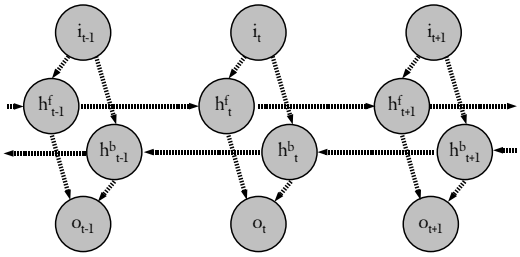


Fig. 2. Structure of a bidirectional network with input i, output o, and two hidden layers (h^f and h^b) for forward and backward processing.

the speech signal. These spikes are separated by long periods during which the *non-keyword* output unit is activated. In the following, we will briefly introduce the concept of Connectionist Temporal Classification.

A major problem with standard objective functions for RNNs is that they require individual targets for each point in the data sequence, which in turn requires the boundaries between segments with different labels (e. g., the keyword boundaries in speech) to be pre-determined. The Connectionist Temporal Classification output layer [Graves et al. 2006] solves this problem by allowing the network to choose the location as well as the class of each label. By summing up over all sets of label locations that yield the same label sequence, CTC determines a probability distribution over possible labelings, conditioned on the input sequence.

The CTC layer as used by Fernandez et al. [2007a] has as many output units as there are distinct keywords, plus an extra *blank* unit for *garbage* speech. The activations of the outputs at each timestep are normalized and interpreted as the probability of observing the corresponding keyword (or no keyword) at that point in the sequence. Because these probabilities are conditionally independent given the input sequence, the total probability of a given (framewise) sequence $w^f_{1:T}$ of blanks and keywords is

$$p(w^f_{1:T}|x_{1:T}) = \prod_{t=1}^{T} o_t^{w^f_t}, \tag{1}$$

where $x_{1:T}$ is a length T input sequence and o^k_t is the activation of output unit k at time t. In order to sum over all the output sequences corresponding to a particular labeling (regardless of the *location* of the labels) we define an operator $\mathcal{B}(\cdot)$ that removes first the repeated labels and then the blanks from the output sequence so that, e. g., $\mathcal{B}(AA--BBB-B) = ABB$. The total probability of the length V labeling $w_{1:V}$, where $V \leq T$, is then

$$p(w_{1:V}|x_{1:T}) = \sum_{w^f_{1:T}:\mathcal{B}(w^f_{1:T})=w_{1:V}} p(w^f_{1:T}|x_{1:T}). \tag{2}$$

A naive calculation of Equation 2 is unfeasible, because the number of $w^f_{1:T}$ terms corresponding to each labeling increases exponentially with the sequence length.

However, $p(w_{1:V}|x_{1:T})$ can be efficiently calculated with a dynamic programming algorithm similar to the forward-backward algorithm for HMMs (see Graves et al. [2006]).

The CTC objective function O^{CTC} is defined as the negative log likelihood of the training set $\mathbb{S}$

$$O^{CTC} = -\sum_{(x_{1:T},w_{1:V})\in\mathbb{S}} \ln p(w_{1:V}|x_{1:T}). \tag{3}$$

An RNN with a CTC output layer can be trained with gradient descent via back-propagation through time, using the following partial derivatives of O^{CTC} with respect to the output activations:

$$\frac{\partial O^{CTC}}{\partial o_t^k} = \frac{-1}{p(w_{1:V}|x_{1:T})o_t^k} \sum_{v\in lab(w_{1:V},k)} \alpha_t(v)\beta_t(v), \tag{4}$$

where $lab(w_{1:V}, k)$ is the set of positions in $w_{1:V}$ where the label k occurs. $\alpha_t(v)$ and $\beta_t(v)$ denote the forward and backward variables as defined by Graves et al. [2006].

When a new input sequence is presented to a network trained with CTC, the output activations (corresponding to the keyword probabilities) tend to form *spikes* separated by long intervals where the blank label is emitted. The location of the spikes corresponds to the portion of the input sequence where the keyword is detected.

CTC has successfully been applied e.g. to handwriting recognition [Graves et al. 2008b], hierarchical sequence labeling [Fernandez et al. 2007b], and phoneme recognition [Graves 2008].

5. TANDEM BLSTM-DBN MODELING

As an alternative to keyword spotting via CTC, this section introduces our Tandem BLSTM-DBN architecture which models keywords and garbage speech at the *phoneme level* and thus can be used for *vocabulary independent* keyword detection. The BLSTM network as applied in our Tandem model has one output unit for every phoneme which allows the network to output phoneme predictions for every time frame. In addition to conventional MFCC speech features, those phoneme predictions are processed by a Dynamic Bayesian Network designed to detect keywords in continuous speech. In Section 5.1 we summarize the basic principle of DBNs before we introduce the Tandem decoder in Section 5.2. Section 5.3 will explain the DBN we used to train the Tandem model.

5.1 Dynamic Bayesian Networks

Dynamic Bayesian Networks are graphical models which consist of a set of nodes and edges. Nodes represent random variables which can be either hidden or observed. Edges – or rather *missing* edges – encode conditional independence assumptions that are used to determine valid factorizations of the joint probability distribution. Dynamic Bayesian Networks are well-suited for speech recognition tasks, since they consist of repeated template structures over time, modeling the temporal evolution of a speech sequence. Conventional Hidden Markov Model approaches can be inter-

preted as *implicit* graph representations using a single Markov chain together with an integer state to represent all contextual and control information determining the allowable sequencing. In this contribution however, we decided for the *explicit* approach [Bilmes and Bartels 2005], where information such as the current phoneme, the indication of a phoneme transition, or the position within a word is expressed by random variables. As shown by Bilmes and Bartels [2005], explicit graph representations are advantageous whenever the set of hidden variables has factorization constraints or consists of multiple hierarchies.

5.2 Decoding

The Tandem BLSTM-DBN architecture for keyword spotting is depicted in Figure 3. The network is composed of five different layers and hierarchy levels respectively: a word layer, a phoneme layer, a state layer, the observed features, and the BLSTM layer (nodes inside the grey shaded box). As can be seen in Figure 3, the DBN jointly processes speech features and BLSTM phoneme predictions. The BLSTM layer consists of an input layer i_t, two hidden layers h_t^f and h_t^b (one for forward and one for backward processing), and an output layer o_t.

The following random variables are defined for every time step t: q_t denotes the phoneme identity, q_t^{ps} represents the position within the phoneme, q_t^{tr} indicates a phoneme transition, s_t is the current state with s_t^{tr} indicating a state transition, and x_t denotes the observed acoustic features. The variables w_t, w_t^{ps}, and w_t^{tr} are identity, position, and transition variables for the word layer of the DBN whereas a hidden *garbage variable* g_t indicates whether the current word is a keyword or not. A second observed variable b_t contains the phoneme prediction of the BLSTM. A short description of the used random variables can bee found in Table II. Figure 3 displays hidden variables as circles and observed variables as squares. Deterministic relations are represented by straight lines, and zig-zagged lines correspond to random conditional probability functions (CPFs). Dotted lines refer to so-called *switching parents* [Bilmes 2003], which allow a variable's parents to change conditioned on the current value of the switching parent. They can change not only the set of parents but also the implementation (i. e., the CPF) of a parent. The bold dashed lines in the BLSTM layer do not represent statistical relations but simple data streams.

Assuming a speech sequence of length T, the DBN structure specifies the factorization

$$\begin{aligned} &p(g_{1:T}, w_{1:T}, w_{1:T}^{tr}, w_{1:T}^{ps}, q_{1:T}, q_{1:T}^{tr}, q_{1:T}^{ps}, s_{1:T}^{tr}, s_{1:T}, x_{1:T}, b_{1:T}) = \\ &f(q_1^{ps}) p(q_1 | w_1^{ps}, w_1, g_1) f(w_1^{ps}) p(w_1) \prod_{t=1}^{T} p(x_t|s_t) p(b_t|s_t) f(s_t|q_t^{ps}, q_t) p(s_t^{tr}|s_t) \\ &f(q_t^{tr}|q_t^{ps}, q_t, s_t^{tr}) f(w_t^{tr}|q_t^{tr}, w_t^{ps}, w_t) f(g_t|w_t) \prod_{t=2}^{T} f(q_t^{ps}|s_{t-1}^{tr}, q_{t-1}^{ps}, q_{t-1}^{tr}) \\ &p(w_t|w_{t-1}^{tr}, w_{t-1}) p(q_t|q_{t-1}^{tr}, q_{t-1}, w_t^{ps}, w_t, g_t) f(w_t^{ps}|q_{t-1}^{tr}, w_{t-1}^{ps}, w_{t-1}^{tr}) \end{aligned} \tag{5}$$

with $p(\cdot)$ denoting random conditional probability functions and $f(\cdot)$ describing deterministic relations (see Table III for an overview over the individual CPFs).

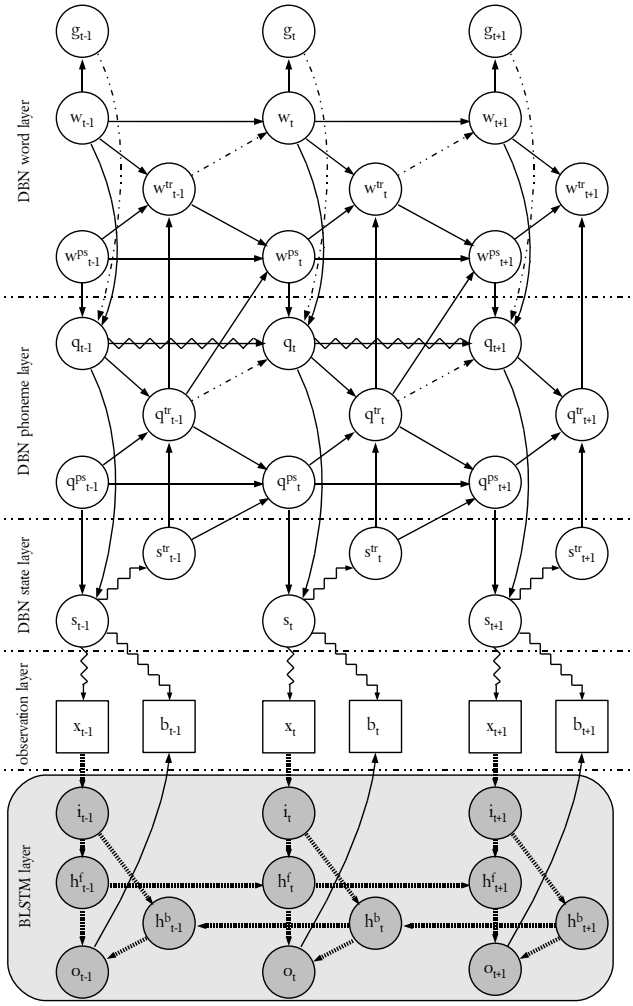


Fig. 3. Structure of the Tandem BLSTM-DBN keyword spotter: The BLSTM network (grey shaded box) provides a discrete phoneme prediction feature b_t which is observed by the DBN, in addition to the MFCC features x_t. The DBN is composed of a state, phoneme, and word layer, consisting of hidden transition (s_t^{tr},q_t^{tr},w_t^{tr}), position (q_t^{ps},w_t^{ps}), and identity (s_t,q_t,w_t) variables. Hidden variables (circles) and observed variables (squares) are connected via random CPFs (zig-zagged lines) or deterministic relations (straight lines). Switching parent dependencies are indicated with dotted lines.

The factorization of Equation 5 can be easily derived when inspecting the DBN layers of Figure 3: In principle we have to build the product of all time steps and all variables while considering that variables might be conditioned on other (parent) variables. This corresponds to arrows in Figure 3 that point to the corresponding (child) node. In case all parent nodes of a child node are located in the same time

frame as the child node, we can build the product from $t = 1$ to $t = T$. Otherwise, if a variable is conditioned on variables from the previous time step, we build the product from $t = 2$ to $t = T$ and define initial CPFs for time step $t = 1$ that are not conditioned on variables from the previous time step (as for example $p(w_1)$). The factorization property in Equation 5 can be exploited to optimally distribute the sums over the hidden variables into the products, using the junction tree algorithm [Jensen 1996]. If S denotes the state space, time and space complexity of the DBN is $\mathcal{O}(ST \log T)$ and $\mathcal{O}(S \log T)$, respectively [Zweig and Padmanabhan 2000].

var.	**meaning**
g_t	garbage variable: equal to 1 if arbitrary speech is decoded; equal to 0 for keywords
w_t	word identity: equal to 0 for garbage speech; equal to $1, \ldots, K$ for keywords
w_t^{tr}	word transition: equal to 1 if a word transition occurs; zero otherwise
w_t^{ps}	word position: position (i. e., phoneme) within a word
q_t	phoneme identity: takes values between 0 and 64 depending on the current phoneme
q_t^{tr}	phoneme transition: equal to 1 if a phoneme transition occurs; zero otherwise
q_t^{ps}	phoneme position: position (i. e., state) within a phoneme
s_t^{tr}	state transition: equal to 1 if a state transition occurs; zero otherwise
s_t	state identity: takes values between 0 and 191 depending on the current state
x_t	vector of continuous MFCC observations
b_t	discrete BLSTM phoneme prediction feature: takes values between 0 and $P - 1$

Table II. Variables used in the DBN.

CPF	**meaning**
$p(w_t \vert w_{t-1}^{tr}, w_{t-1})$	word ID only changes if a word transition occurs (using a priori likelihoods as defined in Equations 7 and 8)
$f(w_t^{tr} \vert q_t^{tr}, w_t^{ps}, w_t)$	word transition occurs in case of a phoneme transition in the last phoneme of a word
$f(w_t^{ps} \vert q_{t-1}^{tr}, w_{t-1}^{ps}, w_{t-1}^{tr})$	word position is incremented if a phoneme transition occurs or set to one if a word transition occurs
$p(q_t \vert q_{t-1}^{tr}, q_{t-1}, w_t^{ps}, w_t, g_t)$	depends on w_t and w_t^{ps} if a keyword is decoded; uses a phoneme bigram otherwise
$f(q_t^{tr} \vert q_t^{ps}, q_t, s_t^{tr})$	phoneme transition occurs in case of a state transition in the last state of a phoneme
$f(q_t^{ps} \vert s_{t-1}^{tr}, q_{t-1}^{ps}, q_{t-1}^{tr})$	phoneme position is incremented if a state transition occurs or set to one if a phoneme transition occurs
$p(s_t^{tr} \vert s_t)$	trained state transition probabilities
$f(s_t \vert q_t^{ps}, q_t)$	mapping from phoneme and phoneme position to state identity
$p(x_t \vert s_t)$	continuous emission probability distribution for MFCC features
$p(b_t \vert s_t)$	discrete probability distribution for BLSTM phoneme prediction

Table III. Deterministic relations and random conditional probability functions (CPF) used in the DBN.

The size of the BLSTM input layer i_t corresponds to the dimensionality of the acoustic feature vector x_t whereas the vector o_t contains one probability score for

each of the P different phonemes at each time step. b_t is the index of the most likely phoneme:

$$b_t = \arg\max_j(o_{t,1}, ..., o_{t,j}, ..., o_{t,P}) \tag{6}$$

The CPFs $p(x_t|s_t)$ are described by Gaussian mixtures as common in an HMM system. Together with $p(b_t|s_t)$ and $p(s_t^{tr}|s_t)$, they are learned via EM training. Thereby s_t^{tr} is a binary variable, indicating whether a state transition takes place or not. Since the current state is known with certainty, given the phoneme and the phoneme position, $f(s_t|q_t^{ps}, q_t)$ is purely deterministic. A phoneme transition occurs whenever $s_t^{tr} = 1$ and $q_t^{ps} = S_q$ provided that S_q denotes the number of states of a phoneme. This is expressed by the function $f(q_t^{tr}|q_t^{ps}, q_t, s_t^{tr})$. The phoneme position q_t^{ps} is known with certainty if s_{t-1}^{tr}, q_{t-1}^{ps}, and q_{t-1}^{tr} are given. The hidden variable w_t can take values in the range $w_t = 0...K$ with K being the number of different keywords in the vocabulary. In case $w_t = 0$ the model is in the *garbage state* which means that no keyword is uttered at that time. Being in the garbage state corresponds to $g_t = 1$.

In our experiments we simplified the word bigram $p(w_t|w_{t-1}^{tr} = 1, w_{t-1})$ to a unigram which makes each keyword equally likely (in order to not favor certain keywords). Yet, we introduced different a priori likelihoods for keywords and garbage phonemes:

$$p(w_t = 1 : K|w_{t-1}^{tr} = 1) = \frac{K \cdot 10^a}{K \cdot 10^a + 1} \tag{7}$$

and

$$p(w_t = 0|w_{t-1}^{tr} = 1) = \frac{1}{K \cdot 10^a + 1}. \tag{8}$$

The parameter a can be used to adjust the trade-off between true positives and false positives. Setting $a = 0$ means that the a priori probability of a keyword and the probability that the current phoneme does not belong to a keyword are equal. Adjusting $a > 0$ implies a more aggressive search for keywords, leading to higher true positive and false positive rates.

As in [Wöllmer et al. 2009d], we assume that 'garbage words' always consist of only one phoneme. The variable q_t has two switching parents: q_{t-1}^{tr} and g_t. Similar to the word layer, q_t is equal to q_{t-1} if $q_{t-1}^{tr} = 0$. Otherwise, the switching parent g_t determines the parents of q_t. In case $g_t = 0$ – meaning that the current word is a keyword – q_t is a deterministic function of the current keyword w_t and the position within the keyword w_t^{ps}. If the model is in the garbage state, q_t only depends on q_{t-1} in a way that phoneme transitions between identical phonemes are forbidden.

Note that the design of the CPF $p(q_t|q_{t-1}^{tr}, q_{t-1}, w_t^{ps}, w_t, g_t)$ entails that the DBN will strongly tend to choose $g_t = 0$ (i. e., it will detect a keyword) once a phoneme sequence that corresponds to a keyword is observed. Decoding such an observation while being in the garbage state $g_t = 1$ would lead to 'phoneme transition penalties' since the CPF $p(q_t|q_{t-1}^{tr} = 1, q_{t-1}, w_t^{ps}, w_t, g_t = 1)$ contains phoneme transition probabilities lower than one. By contrast, $p(q_t|q_{t-1}^{tr} = 1, w_t^{ps}, w_t, g_t = 0)$ is deterministic, introducing no likelihood penalties at phoneme borders.

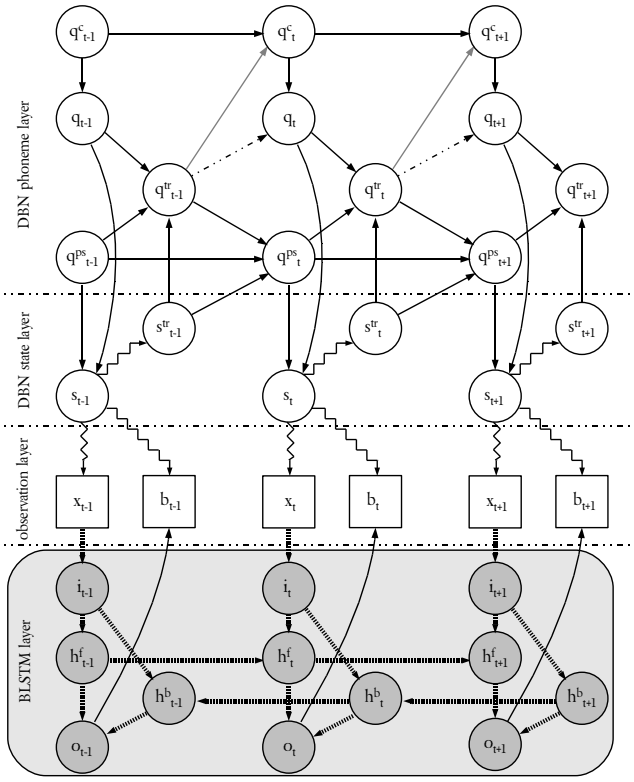


Fig. 4. DBN structure of the graphical model used to train the Tandem keyword spotter: A count variable q_t^c determines the current position in the phoneme sequence.

5.3 Training

The graphical model applied for learning the random CPFs $p(x_t|s_t)$, $p(s_t^{tr}|s_t)$, and $p(b_t|s_t)$ is depicted in Figure 4. Compared to the GM used for keyword decoding (see Section 5.2), the GM for the training of the keyword spotter is less complex, since during (vocabulary independent) training, only phonemes are modeled. The training procedure is split up into two stages: In the first stage phonemes are trained framewise, whereas during the second stage, the segmentation constraints are relaxed using a forced alignment (embedded training).

The variable q_t^c shown in Figure 4 is a count variable determining the current position in the phoneme sequence. Note that the grey-shaded arrow in Figure 4, pointing from q_{t-1}^{tr} to q_t^c is only valid during the second training cycle when there are no segmentation constraints, and will be ignored in Equation 9.

For a training sequence of length T, the DBN structure of Figure 4 specifies the factorization

$$p(q^c_{1:T}, q_{1:T}, q^{tr}_{1:T}, q^{ps}_{1:T}, s^{tr}_{1:T}, s_{1:T}, x_{1:T}, b_{1:T}) = \\ f(q^{ps}_1) f(q^c_1) \prod_{t=1}^{T} p(x_t|s_t) p(b_t|s_t) f(s_t|q^{ps}_t, q_t) p(s^{tr}_t|s_t) f(q^{tr}_t|q^{ps}_t, q_t, s^{tr}_t) f(q_t|q^c_t) \\ \prod_{t=2}^{T} f(q^{ps}_t|s^{tr}_{t-1}, q^{ps}_{t-1}, q^{tr}_{t-1}) f(q^c_t|q^c_{t-1}). \tag{9}$$

During training, the current phoneme q_t is known, given the position q^c_t in the training utterance, which implies a deterministic mapping $f(q_t|q^c_t)$. In the first training cycle q^c_t is incremented in every time frame, whereas in the second cycle q^c_t is only incremented if $q^{tr}_{t-1} = 1$.

6. EXPERIMENTS

We implemented and evaluated five different keyword spotting techniques: the Tandem BLSTM-DBN approach introduced in Section 5, the CTC method as proposed in [Fernandez et al. 2007a], the DBN outlined in [Wöllmer et al. 2009d], a conventional phoneme-based HMM system, and a multi-stream HMM approach that incorporates BLSTM phoneme predictions as an additional discrete stream of observations. Using a set of 25 keywords (see Section 6.1) we will investigate the performance of the respective techniques focussing on the task of keyword detection in a child-robot interaction scenario as outlined in Section 2.

6.1 Keywords

The keyword vocabulary consists of three different categories: words expressing positive valence, words expressing negative valence, and command words (see Table IV). Keywords indicating positive or negative valence were included to allow the Aibo robot to be sensitive to positive or negative feedback from the child. Such keywords can also be used as linguistic features for automatic emotion recognition [Batliner et al. 2006; Steidl 2009; Schuller et al. 2009a; Batliner et al. 2010]. Examples are (German) words like *fein*, *gut*, *böse*, etc. (Engl.: *fine*, *good*, *bad*). Command words like *links*, *rechts*, *hinsetzen*, etc. (Engl.: *left*, *right*, *sit down*) were included so that the children are able to control the Aibo robot via speech. The dictionary contains multiple pronunciation variants as well as multiple forms of the (lemmatized) keywords listed in Table IV. For example the word *umdrehen* (Engl.: *turn around*) can also be pronounced as *umdrehn* and verbs do not necessarily have to be uttered in the infinitive form (e. g., *gehen* (Engl.: *go*) can also be *geh*, *gehst*, or *geht*). In order to allow a fair comparison between techniques that depend on frequent keyword occurrences in the training set (such as the CTC method) and our vocabulary independent approach, only those command words or emotionally relevant words that occurred at least 50 times (incl. variants) in the FAU Aibo Emotion Corpus were included in the vocabulary. In total, there are 82 different entries in the dictionary which are mapped onto exactly 25 keywords as listed in Table IV.

In the test set, 85.6 % of the turns contain at least one keyword; 40.6 % of the turns contain two or more keywords. The average number of keywords contained

category	German keywords	translation
positive valence	brav, fein, gut, schön	well-behaved/good, fine, good, nice
negative valence	böse, nein, nicht	bad, no, not
commands	aufstehen, bleiben, drehen, gehen, geradeaus, hinsetzen, kommen, laufen, links, rechts, setzen, stehen, stehenbleiben, stellen, stopp, tanzen, umdrehen, weiterlaufen	stand up, keep, turn, go, straight, sit down, come, run, left, right, sit, stand, stand still, put, stop, dance, turn around, keep running

Table IV. Keywords

in a turn is 1.4 and the average number of words per turn is 3.6.

6.2 Tandem BLSTM-DBN Training

As for all our experiments, the acoustic feature vectors used for the Tandem BLSTM-DBN keyword spotter consisted of cepstral mean normalized MFCC coefficients 1 to 12, log. energy, as well as first and second order delta coefficients. The BLSTM network was trained on the *framewise* phoneme segmentations of the training set. Since the corpus is only transcribed at the word level, we applied an HMM system as described in Section 6.4 in order to obtain the phoneme-level forced alignments. The BLSTM input layer had a size of 39 (one for each feature) and the size of the output layer was 65 since we modeled a set of 54 German phonemes with additional targets for *silence*, *short pause*, *breathing*, *coughing*, *laughing*, *unidentifiable phonemes*, *noise*, *human noise*, *nasal hesitation*, *vocal hesitation*, and *nasal+vocal hesitation*. Both hidden layers (for forward and backward processing) consisted of one backpropagation layer with 65 hidden cells and two LSTM layers with 130 and 65 memory blocks, respectively. Thereby each memory block consisted of one memory cell. Input and output gates used hyperbolic tangent (tanh) activation functions, while the forget gates had logistic activation functions.

The BLSTM network was trained with standard backpropagation through time (BPTT) [Williams and Zipser 1995] using the exact error gradient as in [Graves et al. 2005]. We used a learning rate of 10^{-5}. As a common means to improve generalization for recurrent neural networks, zero mean Gaussian noise with standard deviation 0.6 was added to the inputs during training. Before training, all weights of the BLSTM network were randomly initialized in the range from -0.1 to 0.1. We aborted training as soon as no improvement on the validation set (two *Ohm*-sessions, see Section 2) could be observed for at least 50 epochs, and chose the network that achieved the best framewise phoneme error rate on the validation set. The resulting frame error rate on the test set is 15.1 %. Note that for the BLSTM-DBN system, we used the validation set exclusively to determine a stop criterion for BLSTM training and not to tune parameters such as the number of memory blocks. Instead, BLSTM parameters were chosen according to our past experience with BLSTM-based phoneme prediction [Wöllmer et al. 2009c; Wöllmer et al. 2010a].

The DBN was trained as explained in Section 5.3. During the first training cycle of the DBN, models for phonemes and non-linguistic vocalizations were trained framewisely using the *Ohm*-sessions of the FAU Aibo Emotion Corpus. All Gaus-

sian mixtures were split once the change of the overall log likelihood of the training set became less than 0.02 %. The number of mixtures per state was increased to eight. In the second training cycle segmentation constraints were relaxed, whereas no further mixture splitting was conducted. All models were composed of three hidden states.

For the training of the DBN we use GMTK [Bilmes and Zweig 2002] which in turn uses Expectation Maximization (EM) [Dempster et al. 1977] and Generalized EM (GEM) [Bilmes 2008] training, depending on the parameter sharing currently in use [Bilmes and Zweig 2002]. A detailed description of both strategies can be found in [Bilmes 1997].

6.3 CTC Network Training

In order to compare the performance of our Tandem model to the CTC keyword spotter proposed by Fernandez et al. [2007a], we trained a BLSTM network with CTC output layer consisting of one output node per keyword and an additional output unit for the *non-keyword* event (see Section 4). As for the Tandem model, the BLSTM network consisted of one backpropagation layer and two LSTM layers for each input direction (size 65, 130, and 65, respectively). Network training was conducted exactly in the same way as for the Tandem approach (Section 6.2). The only difference is that the CTC network uses keywords rather than phonemes as targets. Note that this leads to *empty* target sequences for training turns which contain no keywords.

6.4 Baseline HMM System

As a baseline experiment, the performance of a phoneme-based keyword spotter using conventional HMM modeling was evaluated. Analogous to the DBN, each of the 54 phonemes was represented by three states (left-to-right HMMs) with eight Gaussian mixtures. Increasing the number of mixture components to more than eight did not result in better recognition accuracies. HMMs for non-linguistic events (see Section 6.2) consisted of nine states. We used cross-word triphone models in order to account for contextual information. The HMMs were trained and optimized using HTK [Young et al. 2006].

For HMM-based keyword detection we defined a set of keyword models and a garbage model. The keyword models estimate the likelihood of a feature vector sequence, given that it corresponds to the keyword phoneme sequence. The garbage model is composed of phoneme HMMs that are fully connected to each other, meaning that it can model any phoneme sequence. Via Viterbi decoding the best path through all models is found, and a keyword is detected as soon as the path passes through the corresponding keyword HMM. In order to be able to adjust the operating point on the Receiver Operating Characteristic (ROC) curve we introduced different a priori likelihoods for keyword and garbage HMMs, identical to the word unigram used for the DBN. Apart from the transition probabilities implied by the unigram, the HMM system uses no additional likelihood penalties at the phoneme borders.

6.5 Multi-Stream BLSTM-HMM System

To investigate the performance gain when including the discrete BLSTM phoneme prediction b_t as an additional feature in the HMM framework described in Section 6.4, we extended the HMM-based system to a multi-stream recognizer modeling MFCC and BLSTM observations in independent feature streams. As for the Tandem BLSTM-DBN approach, MFCC observations are modeled via Gaussian mixtures while the BLSTM feature is modeled using the discrete emission probability distribution $p(b_t|s_t)$. Thus, the BLSTM-HMM system can be interpreted as a combined continuous-discrete multi-stream HMM.

6.6 Results

All five keyword spotting approaches were evaluated on children's speech as contained in the *Mont*-sessions of the FAU Aibo Emotion Corpus. Since only the *Ohm*-sessions are used during training, the experiments are completely speaker-independent. Figure 5 shows a part of the ROC curves displaying the true positive rate (tpr) as a function of the false positive rate (fpr) for the baseline HMM, the multi-stream BLSTM-HMM, the DBN as introduced by Wöllmer et al. [2009d], the CTC method proposed by Fernandez et al. [2007a], as well as for the Tandem BLSTM-DBN. Note that due to the design of the DBN, the full ROC curve – ending at an operating point tpr=1 and fpr=1 – cannot be determined, since the model does not include a confidence threshold that can be set to an arbitrarily low value. The trade-off parameter a for the DBN and the BLSTM-DBN was varied between 0 and 15 (step size 1). Since the CTC framework offers no possibility to adjust the trade-off between a high true positive rate and a low false positive rate, we only get one operating point in the ROC space, corresponding to a true positive rate of 85.2 % at a false positive rate of 0.23 %. This operating point lies almost exactly on the ROC curve of the Tandem BLSTM-DBN so that both techniques can be characterized as equally suited for detecting keywords in the given child-robot interaction scenario. Note, however, that unlike the CTC method, the Tandem approach is more flexible as far as changes in the keyword vocabulary are concerned: As both BLSTM and DBN are phoneme-based, the Tandem model is vocabulary independent. By contrast, the CTC network is trained on whole words, which implies that the whole network would have to be re-trained if a vocabulary entry is to be changed. As discussed in Section 5, the Tandem model also offers a trade-off parameter a which can be increased for a more aggressive keyword search. Thus, if a higher false positive rate can be tolerated, the Tandem approach achieves a keyword detection rate of up to 95.9 %. As can be seen in Figure 5, the Tandem model prevails over the baseline HMM system. Thereby the performance difference is most significant at lower false positive rates: When evaluating the ROC curve at a false positive rate of 0.4 %, the absolute difference in true positive rates is larger than 12 %. This indicates that for our children's speech scenario, modeling context via Long Short-Term Memory leads to better results than conventional triphone modeling. In general, considering contextual information during decoding seems to be essential, since the DBN approach which models only monophones leads to a lower ROC performance when compared to the triphone HMM system and to systems applying LSTM. At lower false positive rates, modeling the coarticulation

properties of children's speech by applying the principle of Long Short-Term Memory also boosts the performance of the HMM approach which can be seen in the ROC curve for the multi-stream BLSTM-HMM. Yet, the overall performance is slightly better for the Tandem system outlined in Section 5.

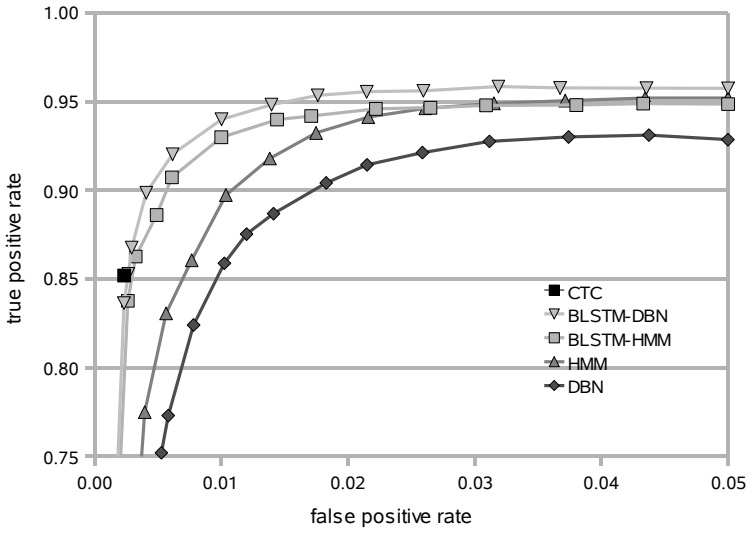
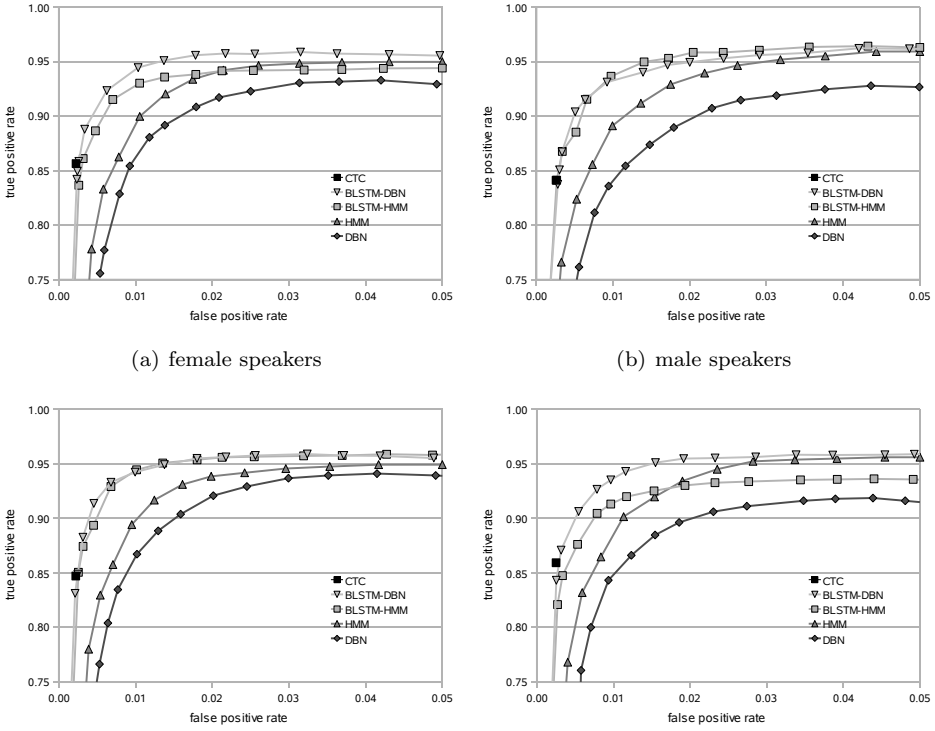


Fig. 5. Evaluation on the FAU Aibo Emotion Corpus (25 keywords): part of the ROC curve for the baseline HMM system, the multi-stream BLSTM-HMM, the DBN keyword spotter (without BLSTM phoneme predictions), the CTC approach, and the Tandem BLSTM-DBN technique. The operating points correspond to $a = 0, 1, 2, 3$, etc. (linear interpolation).

Figures 6(a) to 6(d) show the performance of the five different keyword detection approaches when tested on different fractions of the FAU Aibo Emotion Corpus. Figure 6(a) considers exclusively the 17 female speakers of the *Mont* school while Figure 6(b) shows the word spotting performance for the eight male speakers. For female speakers we can observe a significantly larger performance gap between the multi-stream BLSTM-HMM technique and the Tandem BLSTM-DBN than when considering male speakers, for which both BLSTM-based methods perform almost equally well. Generally, the Tandem approach as proposed in Section 5 prevails over the baseline HMM system for both, female and male speakers – especially at lower false positive rates. Figures 6(c) and 6(d) contain the results for younger (age between 10 and 11 years) and older children (age between 12 and 13 years), respectively. The baseline HMM leads to almost equal performance for both, younger and older children, however, the multi-stream HMM performs significantly better for the younger age group. Again, the Tandem BLSTM-DBN consistently leads to better results when compared to the HMM system, indicating that the Tandem system is suitable for both genders and different age groups. Generally we can observe that the performance of techniques such as the DBN system, the (multi-stream) HMM



(a) female speakers

(b) male speakers

(c) younger children (age 10-11)

(d) older children (age 12-13)

Fig. 6. ROC curves for the different keyword spotting systems evaluated on female speakers, male speakers, younger children (age between 10 and 11 years), and older children (age between 12 and 13 years).

approach, and the CTC method shows a higher dependency on the childrens' age and gender than the proposed Tandem BLSTM-DBN.

Table V shows the average true positive rates for individual keywords at a false positive rate of 1 %. Keywords are grouped into words expressing positive valence, words expressing negative valence, and command words, according to Table IV. For all keyword spotting systems, we observe the same trend: Command words seem to be easier to detect than words related to valence. Besides differences in phonetic composition and lengths of keywords, a plausible reason for this phenomenon is that pronunciations of 'positive' or 'negative' words tend to be emotionally colored while command words are rather pronounced in a neutral or emphatic way. Furthermore, for most recognition engines, words expressing negative valence lead to higher error rates than words associated with positive valence. Since the FAU Aibo Emotion Corpus contains emotion annotations at the word-level, we analyzed which emotions are typically assigned to which keyword. Table VI shows the emotion class distributions for each word category: A considerable percentage of 'positive' and 'negative' keywords are pronounced in a *motherese* (positive valence) and *angry*

	DBN		HMM		BLSTM-HMM		BLSTM-DBN	
true positive rate	mean	std.	mean	std.	mean	std.	mean	std.
positive valence	0.716	0.281	0.724	0.223	0.595	0.317	0.741	0.280
negative valence	0.535	0.264	0.576	0.272	0.702	0.244	0.662	0.213
commands	0.817	0.182	0.858	0.118	0.929	0.051	0.926	0.070
all (unweighted)	0.767	0.222	0.803	0.178	0.848	0.194	0.865	0.166
all (weighted)	0.859		0.897		0.930		0.940	

Table V. True positive rates for the DBN, HMM, BLSTM-HMM, and BLSTM-DBN keyword spotter at a false positive rate of 0.01: mean and standard deviation (std.) of the true positive rates for individual keywords expressing positive/negative valence or command words; weighted and unweighted average true positive rate for the complete set of keywords; 'unweighted' refers to the true positive rate averaged over all keywords while 'weighted' means the average of the true positive rates weighted by the number of occurrences of the individual keywords.

[%]	**angry**	**motherese**	**emphatic**	**neutral**
positive valence	0	23	0	77
negative valence	15	0	16	69
commands	4	1	9	86
all	4	2	9	85

Table VI. Emotions assigned to the keyword categories in %: angry, motherese, emphatic, and neutral.

(negative valence) way, respectively, whereas most of the command words are annotated as *neutral* or *emphatic*. Similar results were observed by Schuller et al. [2009a], where emotional children's speech led to higher error rates.

7. CONCLUSION

We proposed and evaluated a Tandem BLSTM-DBN technique tailored for robust keyword detection in a child-robot interaction scenario. Applying the principle of Long Short-Term Memory, our approach allows for flexible coarticulation modeling in children's speech. The system is trained on spontaneous emotionally colored German children's speech recorded during interaction with Sony's pet robot Aibo. Our Tandem recognizer consists of two main components: a bidirectional Long Short-Term Memory recurrent neural network for context-sensitive phoneme prediction and a Dynamic Bayesian Network for detecting keywords in continuous speech. The system is vocabulary independent since it does not apply whole-word models but rather interprets keywords as phoneme sequences. The comparison of the Tandem approach with other state-of-the-art keyword spotting techniques shows that the BLSTM-DBN can achieve the same performance as a recently proposed Connectionist Temporal Classification approach, which however is less flexible since it is based on whole-word modeling. Furthermore, our technique outperforms an HMM system that is based on triphone modeling rather than Long Short-Term Memory: At equal false positive rates, a true positive rate improvement of up to 12 % (absolute) can be achieved. Generally, emotionally colored speech leads to higher error rates than neutral speech.

Our experiments demonstrated that the proposed Tandem technique is equally well suited for female and male children and that the word spotting performance of

the Tandem BLSTM-DBN shows no dependency on the age of the children, while other approaches lead to larger variations of the ROC curves for different age groups and genders. Coarticulation modeling via bidirectional Long Short-Term Memory was shown to increase recognition performance when compared to pure triphone or monophone modeling – especially for younger children who tend to show more variability in their speech production.

To further improve keyword spotting performance in real-life child-robot interaction, the proposed Tandem system can be extended and optimized in the same way as common HMM techniques: Instead of conventional cepstral mean normalized MFCC features, as used in this study, future experiments could incorporate, e. g., model-based feature enhancement techniques [Wöllmer et al. 2010b], vocal tract length normalization, feature transformation via linear discriminant analysis, or histogram equalization [de la Torre et al. 2005]. Furthermore, discriminative training of the emission probability distribution $p(x_t|s_t)$ or model adaptation might result in additional performance gains.

Future research should also focus on the investigation of alternative BLSTM network topologies such as bottleneck architectures [Grezl and Fousek 2008] and on tuning the stream weights for MFCC and BLSTM observations. A further interesting approach towards better recognition performance through combined BLSTM and DBN modeling would be to jointly decode speech with LSTM networks and DBNs by using techniques for data fusion of potentially asynchronous sequences such as multidimensional dynamic time warping [Wöllmer et al. 2009a] or asynchronous Hidden Markov Models [Bengio 2003].

In order to analyze and understand coarticulation effects in children's speech on the one hand and the degree of context modeled by LSTM networks that are trained on children's speech on the other hand, it might be interesting to examine the sequential Jacobian [Graves 2008], i. e., the influence of past RNN inputs on the output at a given time step in the phoneme sequence.

REFERENCES

Batliner, A., Steidl, S., Schuller, B., Seppi, D., Laskowski, K., Vogt, T., Devillers, L., Vidrascu, L., Amir, N., Kessous, L., and Aharonson, V. 2006. Combining efforts for improving automatic classification of emotional user states. In *Proc. of the 5th Slovenian and 1st International Language Technologies Conference*. Ljubljana, Slovenia, 240–245.

Batliner, A., Steidl, S., Schuller, B., Seppi, D., Vogt, T., Wagner, J., Devillers, L., Vidrascu, L., Aharonson, V., and Amir, N. 2010. Whodunnit - searching for the most important feature types signalling emotional user states in speech. *Computer Speech and Language, Special Issue on Affective Speech in real-life interactions*.

Benayed, Y., Fohr, D., Haton, J. P., and Chollet, G. 2003. Confidence measure for keyword spotting using support vector machines. In *Proc. of ICASSP*. Hong Kong, 588–591.

Bengio, S. 2003. An asynchronous hidden markov model for audio-visual speech recognition. *Advances in NIPS 15*, 1–8.

Bengio, Y. 1999. Markovian models for sequential data. *Neural Computing Surveys 2*, 129–162.

Bengio, Y., Simard, P., and Frasconi, P. 1994. Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks 5,* 2, 157–166.

Bilmes, J. A. 1997. A gentle tutorial on the EM algorithm and its application to parameter estimation for gaussian mixture and hidden markov models. Tech. rep., University of Berkeley. Technical Report ICSI-TR-97-02.

Bilmes, J. A. 1998. Maximum mutual information based reduction strategies for cross-correlation based joint distributional modeling. In *Proc. of ICASSP*. Seattle, Washington, 469–472.

Bilmes, J. A. 2003. Graphical models and automatic speech recognition. In *Mathematical Foundations of Speech and Language Processing*, R. Rosenfeld, M. Ostendorf, S. Khudanpur, and M. Johnson, Eds. Springer Verlag, New York, 191–246.

Bilmes, J. A. 2008. Gaussian models in automatic speech recognition. In *Signal Processing in Acoustics*. Springer, New York, 521–555.

Bilmes, J. A. and Bartels, C. 2005. Graphical model architectures for speech recognition. *IEEE Signal Processing Magazine 22,* 5, 89–100.

Bilmes, J. A. and Zweig, G. 2002. The graphical models toolkit: an open source software system for speech and time-series processing. In *Proc. of ICASSP*. Orlando, Florida, 3916–3919.

Boulard, H. and Morgan, N. 1994. *Connectionist speech recognition: A hybrid approach*. Kluwer Adacemic Publishers.

Cortes, C. and Vapnik, V. 1995. Support-vector networks. *Machine Learning 20,* 3, 273–297.

Das, S., Nix, D., and Picheny, M. 1998. Improvements in children's speech recognition performance. In *Proc. of ICASSP*. Seattle, Washington, 433–436.

de la Torre, A., Peinado, A. M., Segura, J. C., Perez-Cordoba, J. L., Benitez, M. C., and Rubio, A. J. 2005. Histogram equalization of speech representation for robust speech recognition. *IEEE Transactions on Speech and Audio Processing 13,* 3, 355–366.

Dempster, A. P., Laird, N. M., and Rubin, D. B. 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of Royal Statistical Society Series B 39*, 185–197.

Ellis, D. P. W., Singh, R., and Sivadas, S. 2001. Tandem acoustic modeling in large-vocabulary recognition. In *Proc. of ICASSP*. Salt Lake City, UT, USA, 517–520.

Fernandez, S., Graves, A., and Schmidhuber, J. 2007a. An application of recurrent neural networks to discriminative keyword spotting. In *Proc. of ICANN*. Porto, Portugal, 220–229.

Fernandez, S., Graves, A., and Schmidhuber, J. 2007b. Sequence labelling in structured domains with hierarchical recurrent neural networks. In *Proc. of IJCAI*. Hyderabad, India.

Foote, J. 1999. An overview of audio information retrieval. *Multimedia Systems 7,* 1, 2–10.

Gerosa, M., Lee, S., Giuliani, D., and Narayanan, S. 2006. Analyzing children's speech: an acoustic study of consonants and consonant-vowel transition. In *Proc. of ICASSP*. Toulouse, France, 393–396.

Giuliani, D. and Gerosa, M. 2003. Investigating recognition of children's speech. In *Proc. of ICASSP*. Hong Kong, 137–140.

Graves, A. 2008. Supervised sequence labelling with recurrent neural networks. Ph.D. thesis, Technische Universität München.

Graves, A., Fernandez, S., Gomez, F., and Schmidhuber, J. 2006. Connectionist temporal classification: Labelling unsegmented data with recurrent neural networks. In *Proc. of ICML*. Pittsburgh, USA, 369–376.

Graves, A., Fernandez, S., Liwicki, M., Bunke, H., and Schmidhuber, J. 2008a. Unconstrained online handwriting recognition with recurrent neural networks. *Advances in Neural Information Processing Systems 20*, 1–8.

Graves, A., Fernandez, S., and Schmidhuber, J. 2005. Bidirectional LSTM networks for improved phoneme classification and recognition. In *Proc. of ICANN*. Warsaw, Poland, 602–610.

Graves, A., Liwicki, M., Fernandez, S., Bertolami, R., Bunke, H., and Schmidhuber, J. 2008b. A novel connectionist system for unconstrained handwriting recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Graves, A. and Schmidhuber, J. 2005. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks 18,* 5-6, 602–610.

Grezl, F. and Fousek, P. 2008. Optimizing bottle-neck features for LVCSR. In *Proc. of ICASSP*. Las Vegas, NV, 4729–4732.

Gustafson, J. and Sjölander, K. 2002. Voice transformations for improving children's speech recognition in a publicly available dialogue system. In *Proc. of ICSLP*. Denver, Colorado, 297–300.

Hagen, A., Pellom, B., and Cole, R. 2007. Highly accurate children's speech recognition for interactive reading tutors using sub-word units. *Speech Communication 49,* 12, 861–873.

Hermansky, H., Ellis, D. P. W., and Sharma, S. 2000. Tandem connectionist feature extraction for conventional HMM systems. In *Proc. of ICASSP*. Istanbul, Turkey, 1635–1638.

Hermansky, H. and Fousek, P. 2008. Multi-resolution RASTA filtering for TANDEM-based ASR. In *Proc. of European Conf. on Speech Communication and Technology*. Lisbon, Portugal, 361–364.

Hochreiter, S., Bengio, Y., Frasconi, P., and Schmidhuber, J. 2001. Gradient flow in recurrent nets: the difficulty of learning long-term dependencies. In *A Field Guide to Dynamical Recurrent Neural Networks*, S. C. Kremer and J. F. Kolen, Eds. IEEE Press, 1–15.

Hochreiter, S. and Schmidhuber, J. 1997. Long short-term memory. *Neural Computation 9,* 8, 1735–1780.

Jaeger, H. 2001. The echo state approach to analyzing and training recurrent neural networks. Tech. rep., Bremen: German National Research Center for Information Technology. (Tech. Rep. No. 148).

Jensen, F. V. 1996. *An introduction to Bayesian Networks*. Springer.

Keshet, J., Grangier, D., and Bengio, S. 2007. Discriminative keyword spotting. In *Proc. of NOLISP*. Paris, France, 47–50.

Ketabdar, H. and Bourlard, H. 2008. Enhanced phone posteriors for improving speech recognition systems. In *IDIAP-RR*. Number 39. 1–23.

Ketabdar, H., Vepa, J., Bengio, S., and Boulard, H. 2006. Posterior based keyword spotting with a priori thresholds. In *IDAIP-RR*. 1–8.

Lang, K. J., Waibel, A. H., and Hinton, G. E. 1990. A time-delay neural network architecture for isolated word recognition. *Neural Networks 3,* 1, 23–43.

Lin, T., Horne, B. G., Tino, P., and Giles, C. L. 1996. Learning long-term dependencies in NARX recurrent neural networks. *IEEE Transactions on Neural Networks 7,* 6, 1329–1338.

Liwicki, M., Graves, A., Fernandez, S., Bunke, H., and Schmidhuber, J. 2007. A novel approach to on-line handwriting recognition based on bidirectional long short-term memory networks. In *Proc. of ICDAR*. Curitiba, Brazil, 367–371.

Mamou, J., Ramabhadran, B., and Siohan, O. 2007. Vocabulary independent spoken term detection. In *Proc. of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*. Amsterdam, The Netherlands, 615–622.

Mayo, C., Scobbie, J. M., Hewlett, N., and Waters, D. 2003. The influence of phonemic awareness development on acoustic cue weighting strategies in children's speech perception. *Journal of Speech, Language, and Hearing Research 46*, 1184–1196.

Murphy, K. 2002. Dynamic bayesian networks: representation, inference and learning. Ph.D. thesis, Dept. EECS, CS Division, Univ. California, Berkeley.

Narayanan, S. and Potamianos, A. 2002. Creating conversational interfaces for children. *IEEE Transactions on Speech and Audio Processing 10,* 2, 65–78.

Potamianos, A., Narayanan, S., and Lee, S. 1997. Automatic speech recognition for children. In *Proc. of Eurospeech*. Rhodes, Greece, 2371–2374.

Quattoni, A., Wang, S., Morency, L. P., Collins, M., and Darrell, T. 2007. Hidden conditional random fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence 29*, 1848–1853.

Repp, B. H. 1986. Some observations on the development of anticipatory coarticulation. *Journal of the Acoustic Society of America 79,* 5, 1616–1619.

Rose, R. C. 1995. Keyword detection in conversational speech utterances using hidden markov model based continuous speech recognition. *Computer Speech and Language 9,* 4, 309–333.

Rose, R. C. and Paul, D. B. 1990. A hidden markov model based keyword recognition system. In *Proc. of ICASSP*. Albuquerque, NM, USA, 129–132.

Schaefer, A. M., Udluft, S., and Zimmermann, H. G. 2008. Learning long-term dependencies with recurrent neural networks. *Neurocomputing 71,* 13-15, 2481–2488.

Schmidhuber, J. 1992. Learning complex extended sequences using the principle of history compression. *Neural Computing 4,* 2, 234–242.

Schröder, M., Cowie, R., Heylen, D., Pantic, M., Pelachaud, C., and Schuller, B. 2008. Towards responsive sensitive artificial listeners. In *Proc. of 4th Intern. Workshop on Human-Computer Conversation*. Bellagio, Italy, 1–6.

Schuller, B., Batliner, A., Steidl, S., and Seppi, D. 2008. Does affect affect automatic recognition of children's speech? In *Proc. of 1st Workshop on Child, Computer and Interaction*. Chania, Crete, Greece.

Schuller, B., Batliner, A., Steidl, S., and Seppi, D. 2009a. Emotion recognition from speech: Putting ASR in the loop. In *Proc. of ICASSP*. Taipei, Taiwan, 4585–4588.

Schuller, B., Müller, R., Eyben, F., Gast, J., Hörnler, B., Wöllmer, M., Rigoll, G., Höthker, A., and Konosu, H. 2009b. Being bored? recognising natural interest by extensive audiovisual integration for real-life application. *Image and Vision Computing Journal (IMAVIS), Special Issue on Visual and Multimodal Analysis of Human Spontaneous Behavior 27,* 12, 1760–1774.

Schuller, B. and Rigoll, G. 2009. Recognising interest in conversational speech - comparing bag of frames and supra-segmental features. In *Proc. of Interspeech*. Brighton, UK, 1999–2002.

Schuster, M. and Paliwal, K. K. 1997. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing 45*, 2673–2681.

Steidl, S. 2009. *Automatic Classification of Emotion-Related User States in Spontaneous Speech*. Logos, Berlin, Germany.

Steidl, S., Batliner, A., Seppi, D., and Schuller, B. 2010. On the impact of children's emotional speech on acoustic and language models. *EURASIP Journal on Audio, Speech, and Music Processing (JASMP), Special Issue on Atypical Speech*. Article ID 783954.

Williams, R. J. and Zipser, D. 1995. Gradient-based learning algorithms for recurrent neural networks and their computational complexity. In *Back-propagation: Theory, Architectures and Applications*, Y. Chauvin and D. E. Rumelhart, Eds. Lawrence Erlbaum Publishers, Hillsdale, N.J., 433–486.

Wöllmer, M., Al-Hames, M., Eyben, F., Schuller, B., and Rigoll, G. 2009a. A multidimensional dynamic time warping algorithm for efficient multimodal fusion of asynchronous data streams. *Neurocomputing 73,* 1-3, 366–380.

Wöllmer, M., Eyben, F., Graves, A., Schuller, B., and Rigoll, G. 2009b. A Tandem BLSTM-DBN architecture for keyword spotting with enhanced context modeling. In *Proc. of NOLISP*. Vic, Spain.

Wöllmer, M., Eyben, F., Keshet, J., Graves, A., Schuller, B., and Rigoll, G. 2009c. Robust discriminative keyword spotting for emotionally colored spontaneous speech using bidirectional LSTM networks. In *Proc. of ICASSP*. Taipei, Taiwan, 3949–3952.

Wöllmer, M., Eyben, F., Schuller, B., and Rigoll, G. 2009d. Robust vocabulary independent keyword spotting with graphical models. In *Proc. of ASRU*. Merano, Italy, 349–353.

Wöllmer, M., Eyben, F., Schuller, B., and Rigoll, G. 2010a. Recognition of spontaneous conversational speech using long short-term memory phoneme predictions. In *Proc. of Interspeech*. Makuhari, Japan, 1946–1949.

Wöllmer, M., Eyben, F., Schuller, B., Sun, Y., Moosmayr, T., and Nguyen-Thien, N. 2009e. Robust in-car spelling recognition - a tandem BLSTM-HMM approach. In *Proc. of Interspeech*. Brighton, UK, 2507–2510.

Wöllmer, M., Klebert, N., and Schuller, B. 2010b. Switching linear dynamic models for recognition of emotionally colored and noisy speech. In *Proc. of ITG*. Bochum, Germany.

Wöllmer, M., Schuller, B., Eyben, F., and Rigoll, G. 2010c. Combining long short-term memory and dynamic bayesian networks for incremental emotion-sensitive artificial listening. *IEEE Journal of Selected Topics in Signal Processing 4,* 5, 867–881.

Yang, H. H., Sharma, S., van Vuuren, S., and Hermansky, H. 2000. Relevance of time-frequency features for phonetic and speaker/channel classification. *Speech Communication 31*, 35–50.

Young, S., Evermann, G., Gales, M., Hain, T., Kershaw, D., Liu, X., Moore, G., Odell, J., Ollason, D., Povey, D., Valtchev, V., and Woodland, P. 2006. *The HTK book (v3.4)*. Cambridge University Press.

Zweig, G. and Padmanabhan, M. 2000. Exact alpha-beta computation in logarithmic space with application to map word graph construction. In *Proc. of ICSLP*. Beijing, China, 855–858.