Learning the Inter-frame Distance for Discriminative Template-based Keyword Detection

David Grangier[1], *Samy Bengio*[2]

[1]IDIAP Research Institute, Martigny, Switzerland
[2]Google Inc., Mountain View, USA

grangier@idiap.ch, bengio@google.com

Abstract

This paper proposes a discriminative approach to template-based keyword detection. We introduce a method to learn the distance used to compare acoustic frames, a crucial element for template matching approaches. The proposed algorithm estimates the distance from data, with the objective to produce a detector maximizing the Area Under the receiver operating Curve (AUC), i.e. the standard evaluation measure for the keyword detection problem. The experiments performed over a large corpus, SpeechDatII, suggest that our model is effective compared to an HMM system, e.g. the proposed approach reaches 93.8% of averaged AUC compared to 87.9% for the HMM.

Index Terms: spoken keyword detection, template matching, discriminative learning, distance learning

1. Introduction

A reliable detection of spoken keywords is required in several application domains. For instance, voice-enabled devices should detect utterances of keywords corresponding to system commands. Other applications include dialog systems, voice mail categorization or spoken document retrieval.

To address the problem of detecting keywords, two alternative strategies are generally adopted: approaches based on Hidden Markov Models (HMMs) [1, 2, 3] or approaches based on Template Matching (TM) [4]. Each strategy has its own advantages and drawbacks. In the case of HMMs, the main advantage lies in the use of a phonetic approach, allowing HMMs to benefit from large amount of annotated speech to build robust acoustic models. On the other hand, HMMs are known for poorly modeling long temporal dependencies, which can only be circumvented with refined features or adaptation techniques [5, 6]. In the case of TM, the main advantage is precisely the use of long temporal context: all the frames of the keyword template, as well as the information about their relative position, are used during the Dynamic Time Warping (DTW) procedure. This provides an implicit modeling of co-articulation effects or speaker dependencies [7]. On the other hand, most TM approaches fail to take advantage of large amount of training data.

This main limitation of TM-based approaches certainly explains the empirical advantage of HMM-based keyword spotters. However, recently, several researchers have worked on template-based approaches for ASR that could benefit from available training data. For instance, in [8], the authors propose to perform template based ASR, relying on data-driven features. Another example can be found in [9], where the authors propose to rely on training data to infer the distance metric used for frame comparison in their template-based approach. In fact, in both cases, the goal is to replace the Euclidean comparison between acoustic feature vectors with a more reliable inter-frame distance. This work builds upon this recent line of research and introduces an approach for learning the inter-frame distance of a TM keyword detector.

For that purpose, we introduce the *Siamese Keyword Identifier* (SKI), a neural network model for inter-frame distance learning. This model adopts a discriminative approach and its parameters are learned to maximize the most common measure used to evaluate keyword detectors: the area under the true-positive versus false-positive curve. The parameterization of SKI is based on Siamese networks, a type of neural network which has shown to be effective for distance learning in the context of computer vision [10]. These choices yield a model that can be efficiently trained over large datasets through *stochastic gradient descent* [11], and which is effective compared to alternative approaches. In fact, SKI has shown to yield an averaged AUC of 93.8% when evaluated on 30 detection tasks over the SpeechDatII corpus [12]. This should be compared to 87.9% and 59.6% for the HMM and the TM baseline respectively.

The remainder of this paper is organized as follows. Section 2 describes our discriminative approach to TM keyword detection. Section 3 presents the experiments performed to assess our model, and compares it to alternative solutions. Finally, Section 4 draws some conclusions and delineates some possible future work.

2. Discriminative Distance Learning for Keyword Detection

This section is divided into three parts. First, we present the generic framework of TM-based keyword detection. Second, we present the most common keyword detector evaluation methodology, and derive a discriminative objective function from it. Finally, we introduce the proposed model, SKI, along with its training algorithm.

2.1. Template-Based Keyword Detection

In the problem of keyword detection, we are given a candidate acoustic sequence $\overline{x}^c = (x_1^c, \dots, x_T^c)$ and a keyword k, and we should determine whether k is present among the words uttered in $\overline{x}^c$. To achieve such a goal, a template-based keyword detector is given a template $\overline{x}^t = (x_1^t, \dots, x_{T'}^t)$, i.e. an acoustic sequence in which k and only k is uttered. The detection is performed according to the distance $D(\overline{x}^c, \overline{x}^t)$ between the sequence $\overline{x}^c$ and the template $\overline{x}^t$: the keyword is considered as detected whenever $D(\overline{x}^c, \overline{x}^t)$ is below a predefined threshold b. The *global* sequence distance $D(\cdot, \cdot)$ is defined from a *local* frame distance $d(\cdot, \cdot)$, relying on a Dynamic Time Warping

(DTW) procedure [13],

$$D(\overline{x}^c, \overline{x}^t) = \min_a \frac{1}{|a|} \sum_{(i,j)\in a} d(x_i^c, x_j^t)$$

where a is an *alignment* between $\overline{x}^c$ and $\overline{x}^t$, i.e. a is a list of index pairs, in which each pair (i, j) aligns a frame x_i^c of $\overline{x}^c$ with a frame x_j^t of $\overline{x}^t$. In other words, a encodes the hypothesized begin and end points of k in $\overline{x}^c$, as well as the local speaking rate variations between sequence $\overline{x}^c$ and $\overline{x}^t$. $|a|$ represents the hypothesized length of k in $\overline{x}^c$ and the corresponding normalization factor prevents biasing towards short alignments. In most cases, the local distance $d(\cdot, \cdot)$ is computed as the Euclidean distance between acoustic features [7]. In the following, we propose a discriminative learning approach to identify a better distance measure from data.

2.2. Discriminative Learning for Keyword Detection

The assessment of a keyword detector is generally based on two quantities, the *False Positive Rate* (FPR) and the *True Positive Rate* (TPR). The FPR measures the percentage of utterances without the keyword which have been misclassified, while the TPR measures the percentage of utterances containing the keyword which have been correctly classified. Given a TM keyword detector, the practitioner should express a trade-off between achieving a high TPR and achieving a low FPR, and selects the detection threshold b accordingly. When such a trade-off is not expressed, the performance of a keyword detector is evaluated with the true positive versus false positive curve, which is obtained by varying the threshold b from the smallest value (no detection) to the largest value (no rejection). In this case, the performance of the system is generally summarized by the Area Under the Curve (AUC). This quantity can be expressed as the *Wilcoxon-Mann-Whitney statistic* [14],

$$\mathrm{AUC}_k = \frac{1}{|R_k||\overline{R}_k|} \sum_{\substack{\overline{x}^+ \in R_k \\ \overline{x}^- \in \overline{R}_k}} \mathrm{I}\{D(\overline{x}^t, \overline{x}^+) < D(\overline{x}^t, \overline{x}^-)\},$$

where $\overline{x}^t$ is the reference template for keyword k, R_k refers to the set of the sequences containing the keyword k, $\overline{R}_k$ refers to the set of the sequences without the keyword, $\mathrm{I}\{\cdot\}$ denotes the indicator function and $|\cdot|$ denotes the cardinality of a set. AUC_k hence estimates the probability that the distance $D(\overline{x}^t, \overline{x}^+)$ assigned to an utterance $\overline{x}^+$ containing the keyword is smaller than the distance $D(\overline{x}^t, \overline{x}^-)$ assigned to an utterance $\overline{x}^-$ without the keyword.

As recently proposed in our work on discriminative learning for phoneme-based keyword spotters [15], a loss function suitable for the maximization of AUC_k can be defined as,

$$L_k = \frac{1}{|R_k||\overline{R}_k|} \sum_{\substack{\overline{x}^+ \in R_k \\ \overline{x}^- \in \overline{R}_k}} l_k(\overline{x}^+, \overline{x}^-),$$

where $l_k(\overline{x}^+, \overline{x}^-) = \max(0, 1 - D(\overline{x}^t, \overline{x}^-) + D(\overline{x}^t, \overline{x}^+))$. One can remark that, $L_k > 1 - \mathrm{AUC}_k$, which implies that the minimization of the L_k yields the maximization of AUC_k.

2.3. A Siamese Network for Inter-Frame Distance Learning

The proposed model, *Siamese Keyword Identifier* (SKI), is based on *Siamese Neural Networks*, a type of neural network introduced to learn distances between images [10]. This approach reformulates the distance learning problem as the problem of identifying a mapping from the input space into an output space, in which the distance would satisfy some desired properties. Adapted to our task, this approach applies the same neural network f_w to the frames of both the candidate sequence $\overline{x}^c$, and the template sequence $\overline{x}^t$, and then computes the L1 distances between the obtained outputs. In other words, the distance between two frames x_i^c, x_j^t is computed as

$$d_w(x_i^c, x_j^t) = |f_w(x_i^c) - f_w(x_i^t)|_1,$$

where f_w is a Multi-Layered Perceptron (MLP) with one hidden layer and parameters w, and $|\cdot|_1$ refers to the L1 norm[1]. Equipped with this inter-frame distance, we can then rewrite the inter-sequence distance as, $\forall\ (\overline{x}^c, \overline{x}^t)$,

$$D_w(\overline{x}^c, \overline{x}^t) = \min_a \frac{1}{|a|} \sum_{(i,j)\in a} d_w(x_i^c, x_j^t),$$

yielding to the architecture depicted on Figure 1. This type of architecture is referred to as *siamese* as one could note that the *same* MLP is duplicated to be applied to the frames of both the candidate and template sequences.

Given this parameterization, our objective is now to select the parameters w such that D_w minimizes the loss L_k. As a function of w, L_k is a composition of differentiable functions with $\min$ and $\max$. Therefore, it belongs to the class of the *generalized differentiable functions* and can be minimized through stochastic gradient descent [16]. This yields an efficient training procedure, which examines the training pairs $(\overline{x}^+, \overline{x}^-) \in R_k \times \overline{R}_k$ one at a time, as shown in Algorithm 1. The random weight initialization procedure used in this algorithm is described in [11]. The learning rate λ, and the number of iterations n are learning hyper-parameters. The other hyper-parameters of the model are the number of hidden units in f_w, and the output dimension of f_w. In our experiments (see Section 3), all hyper-parameters have been selected through validation.

Initialize w randomly
Repeat n **times**
  sample $(\overline{x}^+, \overline{x}^-) \in R_k \times \overline{R}_k$,
  compute gradient of loss for current sample, $\frac{\partial l_k(\overline{x}^+\overline{x}^-)}{\partial w}$,
  update $w \rightarrow w - \lambda \frac{\partial l_k(\overline{x}^+\overline{x}^-)}{\partial w}$

Algorithm 1: Training Procedure

3. Experiments and Results

This section presents the experiments performed to validate our approach. Our experiments are based on the English version of the SpeechDatII corpus [12]. This corpus consists of recorded telephone speech uttered by British speakers. We focus on the task of spotting keywords corresponding to system commands. This task consists of 30 keywords of various length (e.g. add, call, directory, operator, send, etc). Two types of utterances have been used: the sentences labeled as *word spotting phrases with embedded keyword* that generally contain one or more keywords, and the sentences labeled as *phonetically rich sentences*, that generally contain no keywords. This setup yields a to-

[1] The choice of the L1 norm is mainly motivated to ease optimization, refer to [10] for further details.

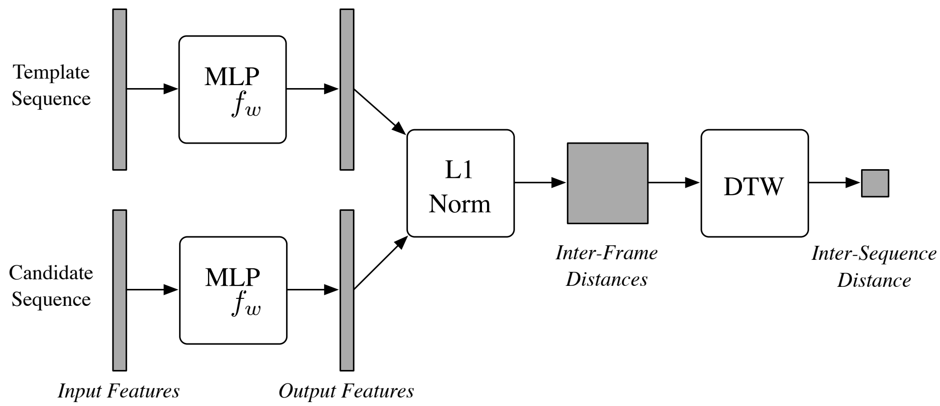

Figure 1: *Architecture of the Model.*

tal set of $10,544$ sequences, which was split into three subset. The training set ($4,758$ sequences) was used to learn the model parameters w, the validation set ($1,000$ sequences) was used to select the model hyper-parameters (see Section 2) and the test set ($4,786$ sequences) was used solely for the purpose of evaluation. The split was performed such that each speaker appears only in one of the sets. The test set provided highly unbalanced detection problems, with a percentage of utterances containing the keyword ranging between 1.5% (dial) and 3.5% (list). All speech sequences were represented using classical Mel Frequency Cepstral Coefficients (MFCCs), with first and second derivatives (Δ and $\Delta\Delta$). Furthermore, Cepstral Mean Subtraction (CMS) was applied to reduce channel variation effects between utterances. We also ran a Gaussian Mixture Model (GMM)-based speech/silence detector in order to shorten long silences at the beginning and end of the utterances [17].

The templates used for our experiments were extracted from the training utterances. For each of the 30 keywords, we randomly selected 10 utterances containing the keyword, and extracted a template from each using the forced alignment data from an HMM/GMM. A detection experiment, including model training and testing, was then performed for each template, and the results were averaged over the template set of each keyword. This prevents biasing the results toward a specific template.

For the sake of comparison, all experiments performed with our model were also conducted relying on a baseline template system and an HMM system. The baseline template system is similar to our model, except that inter-frame comparisons are performed according to the Euclidean distance between MFCC features. This baseline[2] was evaluated with the same templates as those used for the evaluation of our model. The HMM system is composed of 3 emitting states per phoneme, with 50 Gaussians per state. Parameters of the HMM were learned through embedded training over $10,000$ utterances of SpeechDatII, none of those belonging to our test data. Keyword spotting with this HMM is performed through decoding in a model composed of two sub-models, i.e. the keyword sub-model (a left-right HMM connecting the keyword phonemes) and the garbage sub-model (an ergodic HMM connecting all phonemes). With this approach, the keyword is detected whenever the Viterbi best path goes through the keyword sub-model and the trade-off between TPR and FPR is tuned by varying the transition probability leading to the keyword sub-model.

Table 1 reports the average of AUC_k over the 30-keyword

[2]For a more complete evaluation, we are currently planning further comparisons with other template-based approaches, such as [8, 9].

Table 1: *Averaged Area under Curve for the* 30 *Keywords*

	AUC (%)
baseline TM	59.6
HMM	87.9
SKI	93.8

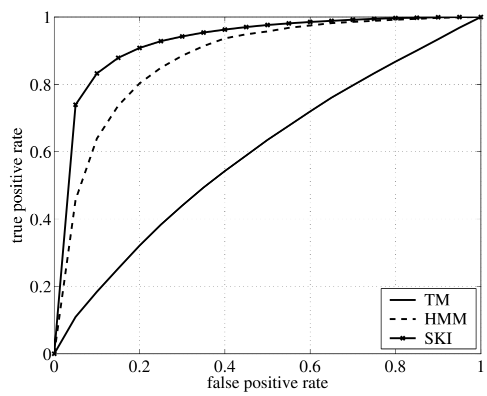

Figure 2: *True Positive vs False Positive Curve (averaged over the* 30 *keyword set)*

set. This table shows that both the HMM and SKI yield good results, compared to the baseline TM. In fact, the baseline TM performs only slightly better than random performance (50% AUC). Compared to the HMM, SKI yields a higher averaged AUC. To verify whether this advantage on the average could be due only to a few keywords, we ran the Wilcoxon test [18] to compare the score of our model with both the TM and the HMM approaches. In both cases, the test rejected this hypothesis at the 95% confidence level, indicating a consistent advantage for SKI. Rather than looking only at the AUC, the practitioner might also be interested at a specific point on the Receiver Operating Curve, depending on his/her system requirements in terms of TPR and FPR. Figure 2 reports the whole curve for the 3 competing models. This plot shows that SKI is actually advantageous over both the TM baseline and the HMM at all operating points.

Analyzing further the results, we report the performance of the HMM and SKI as a function of the keyword length. For that purpose, we grouped the keywords into 3 bins of 10 keywords, according to their average duration in the corpus, and we report the average AUC for each group, see Table 2. For both models, long keywords are better detected than short ones. This seems intuitive as the short words can be easily confused with other

Table 2: *Averaged Area under Curve as a Function of the Keyword Length*

	Avg. Len. (ms)	HMM AUC (%)	SKI (%)
short kw.$^{(a)}$	230 $\rightarrow$ 390	82.5	90.5
medium kw.$^{(b)}$	391 $\rightarrow$ 510	85.1	93.9
long kw.$^{(c)}$	511 $\rightarrow$ 750	95.0	97.0

$^{(a)}$add, delete, dial, end, file, next, play, read, send, stop.
$^{(b)}$call, cancel, change, forward, help, list, record, repeat, reply, save.
$^{(c)}$continue, directory, english, language, menu, operator, previous, program, redial, terminate.

acoustic units, such as part of long words [19]. However, the SKI model seems to be less affected by keyword length compared to the HMM, e.g. the observed drop in performance when comparing short and long keywords is less important for SKI (-6.5%) than for the HMM (-12.5%). In fact, the advantage of SKI is more important for short rather than long keywords, which seems to indicate that template-long context helps to detect the confusing short keywords.

Overall, these results are promising, indicating that template-based approaches can yield competitive keyword detection performance when the inter-frame distance is learned discriminatively.

4. Conclusions

In this paper, we have proposed to improve template-based keyword detection through inter-frame distance learning. The proposed model learns the inter-frame distance from data, with the objective to optimize the area under the true-positive versus false-positive curve of the final detector. An effective online learning strategy has been adopted, allowing the proposed model to be trained over large corpora. We compared our approach over both an HMM-based approach and a simpler template-based approach. Our experiments detecting 30 keywords over the SpeechDatII corpus highlighted the advantage of the proposed model, which yields 93.8% averaged AUC as compared to 59.6% for the baseline template-based approach and 87.9% for the HMM. An explanation for this positive outcome certainly lies in the combined advantage of our approach: like the HMM, the proposed model can benefit from large amount of training data, and, like any template-based approach, our model can also model long temporal dependencies, through the use of a template-long context.

This work opens several possible future directions of research. One of the most promising would be to learn a single model from many templates of different keywords, instead of learning a model per keyword. Such a model could then be applied to detect a new keyword for which only a single template would be given at test time, yielding a model allowing the retrieval of spoken documents from spoken queries.

Acknowledgements The authors would like to thank Joseph Keshet and Johnny Mariéthoz for their help and comments. This work has been performed with the support of the Swiss NSF through the MULTI project. It was also supported by the PASCAL European Network of Excellence, funded by the Swiss OFES. Part of this work was done while Samy Bengio was at IDIAP Research Institute.

5. References

[1] J. R. Rohlicek, P. Jeanrenaud, K. N. H. Gish, B. Musicus, and M. Siu, "Phonetic training and language modeling for word spotting," in *Int. Conf. on Acoustic, Speech and Signal Processing*, 1993.

[2] A. Manos and V. Zue, "Segment-based wordspotter using phonetic filler models," in *Int. Conf. on Acoustic, Speech and Signal Processing*, 1997.

[3] J. Junkawitsch, G. Ruske, and H. Hge, "Efficient methods for detecting keywords in continuous speech," in *European Conf. on Speech Communication and Technology*, 1997.

[4] C. Myers, L. Rabiner, and A. Rosenberg, "An investigation of the use of dynamic time warping for word spotting and connected speech recognition," in *Int. Conf. on Acoustic, Speech and Signal Processing*, 1980.

[5] H. Hermansky, "TRAP-TANDEM: data-driven extraction of temporal features from speech," in *Workshop on Automatic Speech Recognition and Understanding*, 2003.

[6] C. Lee and J. Gauvain, "Speaker adaptation based on map estimation of hmm parameters," in *Int. Conf. on Acoustic, Speech and Signal Processing*, 1993.

[7] M. D. Wachter, K. Demuynck, P. Wambacq, and D. van Compernolle, "A locally weighted distance measure for example-based speech recognition," in *Int. Conf. on Acoustic, Speech and Signal Processing*, 2004.

[8] G. Aradilla, J. Vepa, and H. Bourlard, "Using posterior-based features in template matching for speech recognition," in *Int. Conf. on Spoken Language Processing*, 2006.

[9] M. Matton, M. D. Wachter, D. V. Compernolle, and R. Cools, "Maximum mutual information training of distance measures for template based speech recognition," in *Int. Conf. on Speech and Computer*, 2005.

[10] S. Chopra, R. Hadsell, and Y. LeCun, "Learning a similarity metric discriminatively, with application to face verification," in *Conf. on Computer Vision and Pattern Recognition*, 2005.

[11] Y. LeCun, L. Bottou, G. B. Orr, and K. R. Mueller, "Efficient backprop," in *Neural Networks: Tricks of the Trade*, G. B. Orr and K. R. Mueller, Eds. Springer, 1998, ch. 1.

[12] H. Hoege, C. Draxler, H. van den Heuvel, F. Johansen, E. Sanders, and H. Tropf, "Speechdat multilingual speech databases for teleservices: across the finish line," in *European Conf. on Speech Communication and Technology*, 1999.

[13] H. Ney, "The use of a one-stage dynamic programming algorithm for connected word recognition," *IEEE Trans. on Acoustics, Speech, and Signal Processing*, vol. 32, no. 2, 1984.

[14] C. Cortes and M. Mohri, "Confidence intervals for the area under the ROC curve," in *Neural Information Processing Systems*, 2004.

[15] J. Keshet, D. Grangier, and S. Bengio, "Discriminative keyword spotting," in *Workshop on Non-Linear Speech Processsing*, 2007.

[16] Y. M. Ermoliev and V. I. Norkin, "Stochastic generalized gradient method with application to insurance risk management," International Institute for Applied Systems Analysis, Tech. Rep. 21, 1997.

[17] I. Magrin-Chagnolleau, G. Gravier, and R. Blouet, "Overview of the 2000-2001 ELISA consortium research activities," in *A Speaker Odyssey*, 2001.

[18] J. Rice, *Rice, Mathematical Statistics and Data Analysis*. Duxbury Press, 1995.

[19] M. A. Siegler and R. M. Stern, "On the effect of speech rate in large vocabulary speech recognition systems," in *Int. Conf. on Acoustic, Speech and Signal Processing*, 1995.