Vol.2011-SLP-85 No.6
2011/2/4

Stacked Generalization for Topic Classification of Spoken Inquiries

Rafael TORRES,[†1] Hiromichi KAWANAMI,[†2]
Tomoko MATSUI,[†1] Hiroshi SARUWATARI[†1]
and Kiyohiro SHIKANO[†1]

Stacked generalization is a method that allows to combine outputs of multiple classifiers using a second-level classification, minimizing the generalization error of first-level classifiers and achieving greater predictive accuracy. In a previous work, we compared the performance of different methods in the classification in topics of spoken inquiries in Japanese received by a guidance system, where support vector machines and maximum entropy presented the best performances. In the present work, we employ a stacked generalization scheme that uses predictions of support vector machines and maximum entropy classifiers as input for a second-level classification using linear support vector machines. Experimental results show an improvement in the performance from 94.2% to 94.8% in the classification of automatic speech recognition results from adults and from 88.3% to 89.1% for children, when using stacked generalization in comparison to the individual performance of the methods.

1. Introduction

Stacked generalization, originally proposed by Wolpert in 1992[1)], is a method for minimizing the prediction error of individual classifiers, using a second-level classification to combine the predictions of multiple classifiers. Its success arises from its ability to exploit the diversity in the predictions of individual classifiers, achieving greater predictive accuracy at the second level of classification. Wolpert concluded in his work[1)] that some version of stacked generalization should be used to minimize the generalization error in almost any generalization problem.

In the work of Ting *et al.*[2)], the effectiveness of stacked generalization was demonstrated for combining three different types of learning algorithms: C4.5, Naive Bayes and IB1, using a multi-response linear regression algorithm (MLR), for the classification of datasets from the UCI Repository of machine learning databases. In the work of Sigletos *et al.*[3)], stacked generalization was compared against voting, which does not use a second-level classification but takes in consideration only the prediction of the majority of the classifiers, concluding that while voting was effective in most of the tested domains, stacking was consistently effective in all the tested domains. In both works it was concluded that using output class probabilities rather than class predictions from the base level classifiers leads to better classification performance.

Sill *et al.*[4)] implemented a stacked generalization based technique named feature-weighted linear stacking (FWLS), which was a key component in the solution that awarded them the second place in the Netflix Prize competition carried out in 2009. The objective of the competition was to predict the preferences of customers for various products using the Netflix Prize collaborative filtering dataset. Stacked generalization was also extensively used in the solution of the team that won the first prize, which used a blend of hundreds of different models.

In a previous work[5)], we compared the performance of different methods in the topic classification of spoken inquiries in Japanese received by a guidance system operating in a real environment. The system in mention operates in a public space, receiving daily user requests for information and collecting real data. Topic classification of utterances in this kind of systems is useful to identify which are user's main information needs and to ease the selection of proper answers.

In this previous work[5)], it was shown that the best performances were obtained using support vector machines (SVM) and maximum entropy (ME) as classifiers. SVM has successfully been applied in speech classification[6)–8)], because it is appropriate for sparse high-dimensional feature vectors, and it is also robust against speech recognition errors. ME, in the work of Evanini *et al.*[9)], was shown to outperform five other conventional statistical classifiers in the classification of calls, using user's responses to a prompt from an automated troubleshooting dialog system. It was also shown in our previous work that using characters n-grams as features, rather than word n-grams, yields to improvements in the classification

†1 Graduate school of Information Science, Nara Institute of Science and Technology
†2 Department of Statistical Modeling, The Institute of Statistical Mathematics

performance of the methods.

In the present work, we employ a stacked generalization scheme that uses predictions of SVM and ME classifiers as input for a second-level of classification using linear support vector machines. The remainder of the paper is structured as follows: in Section 2, the speech-oriented guidance system *Takemaru-kun* is described. In Section 3, the first and second-level classifiers, as well as the stacked generalization scheme are briefly explained. Section 4 presents the conducted experiments and their results. Finally, Section 5 presents the conclusions.

2. Speech-Oriented Guidance System *Takemaru-kun*

2.1 Description of the System

The *Takemaru-kun* system[10] (Figure 1) is a real-environment speech-oriented guidance system, placed inside the entrance hall of the Ikoma City North Community Center located in the Prefecture of Nara, Japan. The system has been operating daily from November 2002, providing guidance to visitors regarding the center facilities, services, neighboring sightseeing, weather forecast, news, and about the agent itself, among other information. Users can also activate a Web search feature that allows searching for Web pages over the Internet containing the uttered keywords. This system is also aimed at serving as field test of a speech interface, and to collect actual utterance data.

The system displays an animated agent at the front monitor, which is the mascot character of Ikoma city, *Takemaru-kun*. The interaction with the system follows a one-question-to-one-answer strategy, which fits the purpose of responding simple questions to a large number of users. When a user utters an inquiry, the system responds with a synthesized voice, an agent animation, and displays information or Web pages at the monitor in the back, if required.

Since the *Takemaru-kun* system started operating, the received utterances have been recorded. A database containing over 100K utterances recorded from November 2002 to October 2004 was constructed. The utterances were transcribed and manually labeled, pairing them to specific answers. Information concerning the age group, gender and invalid inputs such as noise, level overflowed shouts and other unclear inputs were also documented. Valid utterances showed to be relatively short, with an average length of 3.65 words per utterance. The

Fig. 1 Speech-oriented guidance system *Takemaru-kun*

answer selection in *Takemaru-kun* system is based on 1-nearest neighbor (1-NN), which classifies an input based on the closest example according to a similarity score. An input utterance is compared to example questions in the database, and the answer paired to the most similar example question is output.

We have heuristically defined 40 topics, grouping questions that are related, using the database constructed during the first two years of operation of the system.

3. Classification with Stacked Generalization

In this work, we employ a stacked generalization scheme that uses predictions of SVM and ME classifiers as input for a second-level classification using linear support vector machines. In this section, the first and second-level classifiers as well as the stacked generalization scheme are briefly explained.

3.1 First-Level Classifiers

We used SVM and ME as first-level classifiers. A brief explanation of both classification methods is presented below.

3.1.1 Support Vector Machine

Support Vector Machine (SVM) tries to find optimal hyperplanes in a feature space that maximize the margin of classification of data from two different classes. For this work, LIBSVM[11] was used to apply SVM. Specifically, we are using C-

support vector classification (C-SVC), which implements soft-margin.

We used bag-of-words (BOW) to represent utterances as vectors, where each component of the vector indicates the frequency of appearance of a word. The length of a vector corresponds to the size of the dictionary that includes every word in the training sample set. We selected a radial basis function (RBF) kernel, because in preliminary experiments it presented slightly better performance than a polynomial kernel for this task. The RBF kernel is defined as:

$$\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp(-\gamma||\boldsymbol{x}_i - \boldsymbol{x}_j||^2), \gamma > 0 \tag{1}$$

where $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ represent sample vectors and γ is a hyper-parameter of the function.

In the problem we are addressing, the amount of samples available for each topic is unbalanced. The SVM primal problem formulation implementing soft-margin for unbalanced amount of samples follows the form:

$$\begin{aligned} \min_{\boldsymbol{w},b,\boldsymbol{\xi}} \quad & \frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + C_+ \sum_{y_i=1} \xi_i + C_- \sum_{y_i=-1} \xi_i \\ \text{subject to} \quad & y_i(\boldsymbol{w}^T\phi(\boldsymbol{x}_i) + b) \geq 1 - \xi_i, \\ & \xi_i \geq 0, i = 1, ..., l. \end{aligned} \tag{2}$$

where $\boldsymbol{x}_i \in R^n, i = 1, ..., l$ indicates a training vector, $y_i \in \{1, -1\}$ a class, and ϕ is the function for mapping the training vectors into feature space. The hyper-parameters C_+ and C_- penalize the sum of the slack variable ξ_i for each class, that allows the margin constraints to be slightly violated. By introducing different hyperparameters C_+ and C_-, the unbalanced amount of data problem, in which SVM parameters are not estimated robustly due to unbalanced amount of training vectors for each class, can be dealt with.

SVM is originally designed for binary classification. We implemented the one-vs-rest approach for multi-class classification, which constructs one binary classifier for each topic, and each one is trained with data from a topic, regarded as positive, and the rest of the topics, regarded as negative. We selected one-vs-rest as in preliminary experiments it presented better performance than one-vs-one for this task.

3.1.2 Maximum Entropy

Maximum Entropy (ME)[12] is a technique for estimating probability distributions from data, and it has been widely used in natural language tasks, including speech classification, where it has shown to outperform other conventional statistical classifiers[9].

As it is expressed in[9], given an utterance consisting of the word sequence w_1^N, the objective of the classifier is to provide the most likely class label $\hat{k}$ from a set of labels K:

$$\hat{k} = \underset{k \in K}{\operatorname{argmax}}\ p(k|w_1^N) \tag{3}$$

where the ME paradigm expresses the probability $p(k|w_1^N)$ as:

$$p(k|w_1^N) = \frac{\exp\left[\sum_w N(w) \log \alpha(k|w)\right]}{\sum_{k'} \exp\left[\sum_w N(w) \log \alpha(k|w)\right]}. \tag{4}$$

Ignoring the terms that are constant with respect to k yields:

$$\hat{k} = \underset{k \in K}{\operatorname{argmax}} \sum_w N(w) \log \alpha(k|w). \tag{5}$$

where $N(w)$ is the frequency of a word in an utterance, and $\alpha(k|w)$ with $\alpha(k|w) \geq 0$ and $\sum_k \alpha(k|w) = 1$ are parameters that depend on a class k and a word w.

We applied ME using the package maxent Ver.2.11[13], which uses the L-BFGS-B algorithm to estimate the parameters. L-BFGS-B is a limited-memory algorithm for solving large nonlinear optimization problems subject to simple bounds on the variables. We also selected the ME model with inequality constraints[14], because in preliminary experiments it presented better performance.

3.2 Second-Level Classifier

We used linear support vector machines as second-level classifier. The explanation of the method is the same as the one presented in section 3.1.1, but it differs in the kernel function. The linear kernel is defined as:

$$\kappa(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{x}_i^T \boldsymbol{x}_j. \tag{6}$$

where $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ represent sample vectors.

3.3 Stacked Generalization Scheme

Stacked generalization was originally proposed by Wolpert in 1992[1)], and it is a method for minimizing the prediction error of individual classifiers, using a second-level classification to combine the predictions of multiple classifiers.

In a specific classification task, different classifiers can be used to produce different models. The first step in the stacked generalization training is to collect the predictions of each first-level classifier to create a new set of data, which contains the true classification and the predictions of each one of the first-level classifiers for each one of the samples of the original data set.

To avoid bias, the first-level models should be trained excluding the samples we want to predict, which can be achieved by using cross-validation. In our implementation we used 10-fold cross-validation. The second step is to use the predictions of the first-level classifiers as new data for training a second-level model.

The flow of the implemented stacked generalization training is presented in Figure 2. The original training was split in ten parts to implement 10-fold cross validation; and for this we made sure that each topic was represented in the same proportion on each one of the resultant sets. Then, we trained the first-level models using these sets, and obtained the predictions of SVM RBF and ME for each one of the samples. Finally, we used these predictions to train the second-level model, using SVM linear. The feature vectors of the samples used to train the second-level model contained the real topic, and the SVM RBF predictions (1 or 0) and ME predictions (probability) for each one of the topics.

4. Experiments

We compared the performance of the stacked generalization scheme described in section 3 with the performance of SVM RBF and ME, in the classification in topics of utterances in Japanese received by the speech-oriented guidance system *Takemaru-kun*. We used character unigrams, bigrams and trigrams as features for the training of first-level models. Optimal hyper-parameter values for SVM were obtained experimentally using a grid search strategy, and were set a posteriori. The experiments and obtained results are detailed below.

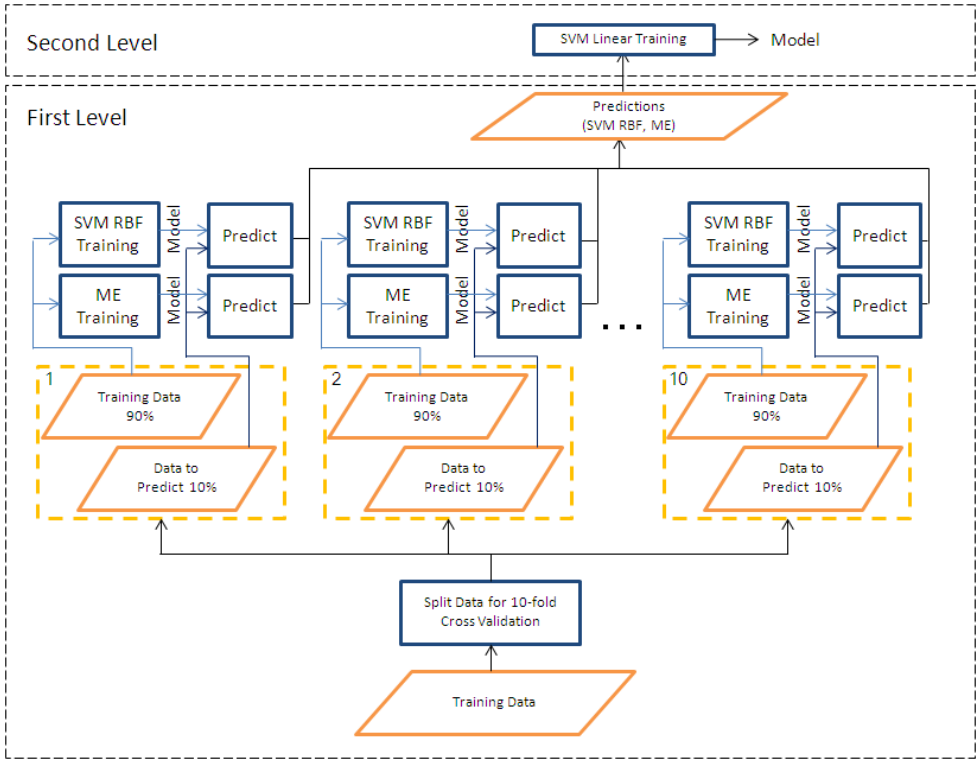

Fig. 2 Stacked generalization flow for training

4.1 Characteristics of the Data

The data used in the experiments was collected by the speech-oriented guidance system *Takemaru-kun* from November 2002 to October 2004, and it was composed by valid utterances from adults and children. Julius Ver.3.5.3 was used as ASR engine. The acoustic model was constructed using the Japanese Newspaper Article Sentences (JNAS) corpus, re-training it with valid samples collected by the system in the period indicated above. The language model was constructed using the transcriptions of the same samples. The samples corresponding to the month of August 2003 were used for the test set and were not included in the training set. The rest of the samples were used for the training set. Table 1

Table 1 ASR word recognition accuracy of the datasets

	Adult		Child	
	Training	Test	Training	Test
ASR Accuracy (%)	85.66	85.10	66.81	67.18

Table 2 Amount of samples per topic

	Adults		Children	
Topic Description	Training	Test	Training	Test
chat-compliments	766	49	2548	200
info-services	494	35	884	88
info-news	484	37	529	65
info-local	553	32	709	65
info-facility	1795	80	5007	423
info-city	504	24	1006	92
info-weather	1099	62	2947	234
info-time	984	53	3911	346
info-sightseeing	668	10	647	28
info-access	676	33	681	59
greeting-end	912	69	4535	437
greeting-start	2672	159	6845	547
agent-name	1309	70	5381	490
agent-likings	851	44	4418	322
agent-age	664	35	3446	342
Total	14431	792	43494	3738

shows the word recognition accuracy of the ASR engine for the datasets.

For these experiments we selected the 15 topics with most training samples. The amount of samples available per topic is shown in Table 2. We conducted experiments with transcriptions and ASR 1-best results.

4.2 Evaluation Criteria

Classification performance of the methods on each topic was evaluated using the F-measure, as defined by:

$$F\text{-}measure = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \tag{7}$$

4.3 Experiment Results

In order to test the second-level model, we obtained predictions of the test set samples by classifying them using first-level models trained with the entire

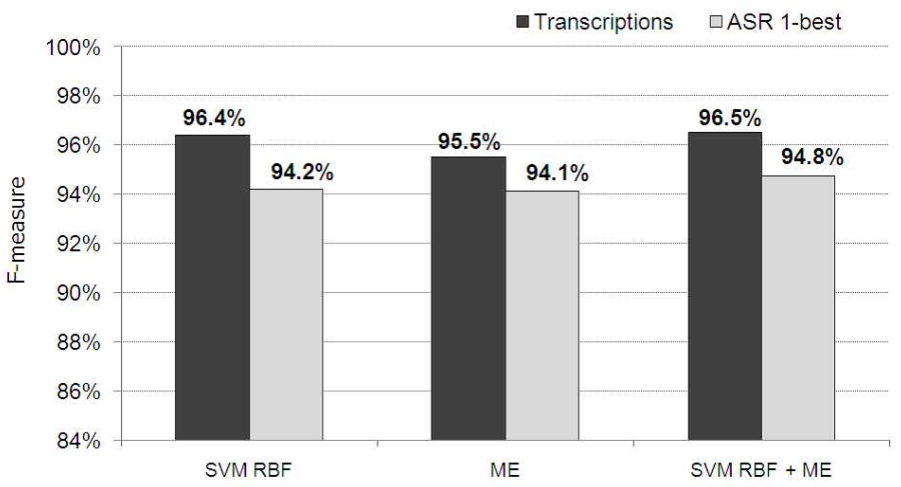

Fig. 3 Best f-measure per method for adults

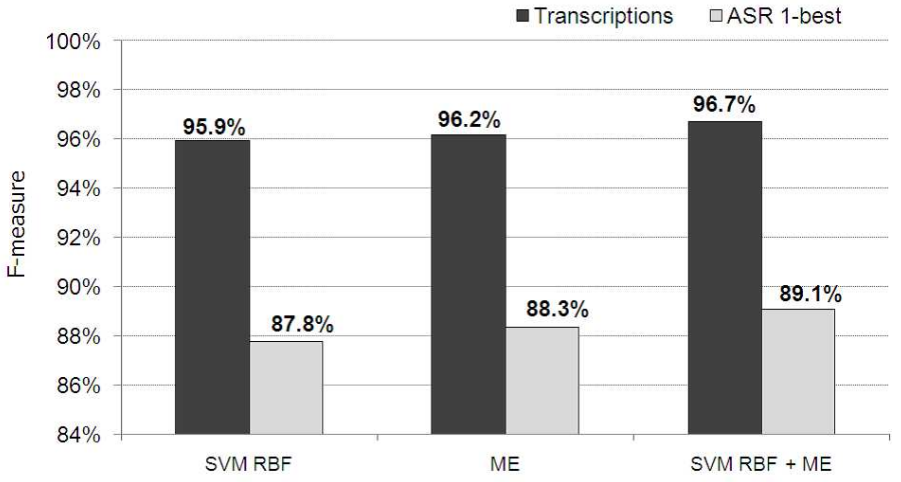

Fig. 4 Best f-measure per method for children

training dataset. Those predictions were used as second-level test data for the second-level model. The classification performance of the second-level model open test was considered the resultant classification performance of the stacking generalization method.

The F-measure was calculated individually for the classification of each topic and it was averaged by frequency of samples in the topics. Figures 3 and 4 present the best performance obtained with SVM RBF and ME individually, and combined using the stacked generalization scheme.

We can observe that in both cases, the classification performance obtained by combining both methods is comparatively higher than the individual performance of the methods. When classifying ASR 1-best results, for adults data SVM RBF presented better performance than ME, with 94.2%, and the combination of both methods yielded to a performance of 94.8%. For children data ME performed better than SVM RBF, with 88.3%, giving a performance of 89.1% when combining both methods.

Experiment results also show a lower classification performance for ASR 1-best results from children, but at the same time this dataset also presented the greatest improvement in comparison to the other datasets.

Additional experiments combining several SVM RBFs classifiers, trained with different hyperparameters, with the ME classifier did not show improvements in classification performance.

5. Conclusions

This work described a stacked generalization scheme to combine predictions of support vector machine and maximum entropy classifiers, using a second-level classification with linear support vector machines. Experimental results show an improvement in the classification performance when combining both methods, in comparison to the performance of the individual classifiers. We could observe that the achieved improvements were relatively small, which is reasonable, as both classifiers presented similar classification errors.

References

1) D.H. Wolpert, "Stacked Generalization," *Neural Networks*, Vol.5(2), pp.241-260, 1992.
2) K.M. Ting, I.H. Witten, "Issues in Stacked Generalization," *Journal of Artificial Intelligence Research*, Vol.10, pp.271-289, 1999.
3) G. Sigletos, G. Paliouras, C.D. Spyropoulos, M. Hatzopoulos, "Combining Information Extraction Systems Using Voting and Stacked Generalization," *Journal of Machine Learning Research*, Vol.6, pp.1751-1782, 2005.
4) J. Sill, G. Takacs, L. Mackey, D. Lin, "Feature-Weighted Linear Stacking," *CoRR*, arXiv:0911.0460, 2009.
5) R. Torres, S. Takeuchi, H. Kawanami, T. Matsui, H. Saruwatari, K. Shikano, "Comparison of Methods for Topic Classification in a Speech-Oriented Guidance System," *In Proc. of Interspeech 2010*, pp.1261-1264, 2010.
6) N. Gupta, G. Tur, D. Hakkani-Tur, S. Bangalore, G. Riccardi, M. Gilbert, "The AT&T Spoken Language Understanding System," *IEEE Trans on. Audio, Speech and Language Processing*, Vol.14, No.1, pp. 213-222, 2006.
7) I. Lane, T. Kawahara, T. Matsui, S. Nakamura, "Out-of-Domain Utterance Detection using Classification Confidences of Multiple Topics," *IEEE Trans. on Speech and Audio Processing*, Vol.15, No.1, pp.150-161, 2007.
8) Y. Park, W. Teiken, S. Gates, "Low-Cost Call Type Classification for Contact Center Calls Using Partial Transcripts," *In Proc. of Interspeech 2009*, pp.2739-2742, 2009.
9) K. Evanini, D. Suendermann, R. Pieraccini, "Call Classification for Automated Troubleshooting on Large Corpora," *In Proc. of ASRU 2007*, pp.207-212, 2007.
10) R. Nisimura, A. Lee, H. Saruwatari, K. Shikano, "Public Speech-Oriented Guidance System with Adult and Child Discrimination Capability," *In Proc. of ICASSP 2004*, Vol.1, pp.433-436, 2004.
11) C. Chang, C. Lin, "LIBSVM: a Library for Support Vector Machines," 2001. Software available at http://www.csie.ntu.edu.tw/˜cjlin/libsvm
12) A. Berger, S. Della Pietra, V. Della Pietra, "A Maximum Entropy Approach to Natural Language Processing," *Computational Linguistics*, Vol.22, No.1, 1996.
13) Maximum Entropy Modeling Package. http://mastarpj.nict.go.jp/mutiyama/software.html#maxent
14) J. Kazama, J. Tsujii, "Evaluation and Extension of Maximum Entropy Models with Inequality Constraints," *In Proc. of EMNLP 2003*, Vol.10, pp.137-144, 2003.