Hierarchical Classification for Speech-to-Speech Translation

Emil Ettelaie, Panayiotis G. Georgiou, Shrikanth S. Narayanan

Signal Analysis and Interpretation Laboratory
Ming Hsieh Department of Electrical Engineering
Viterbi School of Engineering, University of Southern California
3710 S. McClintock Ave., RTH 320, Los Angeles, CA 90089, USA
ettelaie@usc.edu, georgiou@sipi.usc.edu, shri@sipi.usc.edu

Abstract

Concept classifiers have been used in speech to speech translation systems. Their effectiveness, however, depends on the size of the domain that they cover. The main bottleneck in expanding the classifier domain is the degradation in accuracy as the number of classes increase. Here we introduce a hierarchical classification process that aims to scale up the domain without compromising the accuracy. We propose to exploit the categorical associations that naturally appear in the training data to split the domain into sub-domains with fewer classes. We use two methods of language model based classification and topic modeling with latent Dirichlet allocation to use the discourse information for sub-domain detection. The classification task is performed in two steps. First the best category for the discourse is detected using one of the above methods. Then a sub-domain classifier—limited to that category—is deployed. Empirical results from our experiments show higher accuracy for the proposed method compared to a single layered classifier.

Index Terms: utterance classification, topic modeling, latent Dirichlet allocation, speech-to-speech translation

1. Introduction

Concept based classification of spoken utterances has been considered as one of the translation methods in speech to speech (S2S) translation systems [1, 2] since the time of earliest prototypes. Despite its limited domain, this method remains attractive due to its robustness to disfluencies and ASR errors, the quality of its outputs, and the accurate back-translation that it provides to the user especially for structured interactions. Concept classifiers have also been used in other applications involving Spoken Language Understanding (SLU) [3]. Facilitating the exchange of concepts between the interlocutors is usually the design criterion for S2S translation systems. The classifier operates on a finite number of concepts. This simplifies the use of discourse information [4] which can greatly improve the effectiveness of the machine as a concept mediator.

The set of concepts that a classifier handles defines its domain. Since the domain of the classifier is limited it must be accompanied by some other general domain translation engine to cope with the out of domain utterances. In practice the classifier is effective only if its domain is large enough. The accuracy of the classifier, however, decreases as its domain grows because of a larger number of concept classes. In other words, this drop in accuracy is the bottleneck in scaling up the classifier domain.

An intuitive solution to this problem is to divide a large classification domain into smaller sub-domains. In this paper, we introduce a hierarchical classification scheme that exploits a categorical division of the domain. Such divisions naturally exist in almost every domain and are usually the result of the data collection process. For instance in patient care domain, exchange utterances can be grouped in categories such as greeting, medicine dosage instructions, diet, pediatrics, cardiology, and so on. We created the hierarchy by building a category detection layer in to the classifier structure. This can be achieved by using a maximum likelihood classifier implemented by n-gram language models at the category level. Another method is to capture the semantic relations between words and categories through topic modeling. We used Latent Semantic Allocation (LDA) [5] for topic modeling [6].

Document classification has been used as an example in [5] to show the usefulness of topic modeling. In [7], topic distributions were used as additional features in a document classification task, using mined text from the web for topic modeling. Classification of web pages was addressed in [8] by using topic distributions to compute a feature set. Based on that feature set, a Hierarchical Support Vector Machine (HSVM) was trained as the classifier. In contrast to these methods, the focus of our work has been the classification of utterances rather than documents. Specifically, the purpose of this work is to explore a feasible strategy for scaling up the classifier based spoken language processing for S2S applications.

The next section reviews the structure of the classifier that is used for our translation task. In Section 3 we introduce the hierarchical classification scheme based on topic modeling. Section 4 covers the details of experiments performed to test the proposed method along with the discussion of the outcomes. This is followed by conclusions in Section 5.

2. Concept Classifier

From a speech translation point of view we can define a concept class as a group of source language phrases that can all be expressed with the same sentence in the target language. The intent is to convey a conceptual, and not literal, translation. The objective is to use groups of paraphrases to train a classifier that matches an input utterance to the closest concept class. The translation task then reduces to presenting a pre-stored sentence identified with this concept class in the target language.

With the maximum likelihood criterion this process can be expressed as,

$$\hat{c} = \arg\max_{c \in \mathcal{C}} \{P(\mathbf{e} \mid c)\} \quad (1)$$

where $\mathcal{C} \triangleq \{c_1, c_2, \ldots, c_N\}$ is the set of equiprobable concept classes and $\mathbf{e}$ is the input utterance. The likelihood can be approximated by a set of per-class language models (LMs) that are built from the training paraphrases of each class. The classes of-

ten have a very limited vocabulary, therefore the class LMs need interpolation with a large scale background LM, otherwise they would not produce any usable scores. Note that in this work, without loss of generality, we do not consider any priors that may stem from dialog models [4].

With an increase in the number of classes there is a decrease in the discrimination power of the single layer classifier of (1) because of more competing hypotheses. This leads to an increase in the classifier's error rate. Concept classification systems cannot be scaled up unless this problem is addressed.

3. Hierarchical Classifier

3.1. A Two-layered Classifier

The idea of hierarchical classification is to perform the task in different stages. Assume that the domain of a classifier in (1) with N classes is split into K disjoint sub-domains with $N_1, N_2, \ldots, N_K$ classes each, such that $\sum_{i=1}^{K} N_i = N$. We call the classifiers that operate on these sub-domains, sub-classifiers. This division does not change the class LMs. Assume that the original N-ary classifier chooses class $c_{\mathbf{e}}$ for an input utterance $\mathbf{e}$, i.e.,

$$c_{\mathbf{e}} = \arg\max_{c \in \mathcal{C}} \{P_c(\mathbf{e})\} \tag{2}$$

where $P_c(\mathbf{e})$ is the score of $\mathbf{e}$ from the LM of class c. If one selects the sub-classifier with domain $\mathcal{C}_l \subset \mathcal{C}$ such that $c_{\mathbf{e}} \in \mathcal{C}_l$, then,

$$\max_{c \in \mathcal{C}_l} \{P_c(\mathbf{e})\} = \max_{c \in \mathcal{C}} \{P_c(\mathbf{e})\} = P_{c_{\mathbf{e}}}(\mathbf{e}) \tag{3}$$

This means that a sub-classifier will make the same decision as the original classifier as long as its domain contains that decision. This implies that a sub-classifier will not introduce additional errors if 1) its domain contains the correct class and, 2) the original classifier makes a correct decision. Therefore the overall error rate of K sub-classifiers will be less than or equal to the error rate of the flat N-ary classifier, if the right sub-classifier is selected for each input, i.e., if the top level classifier is perfectly accurate. This is supported by the empirical results as we will see in Section 4.

By splitting the original domain into sub-domains we build a two layer classifier. The first layer will select the proper sub-domain, while the second layer will consist of a group of sub-classifiers. For each input, only one of these sub-classifiers will be active based on the first layer selection. The hierarchical classification process will remain accurate as long as each layer of the process remains accurate. Since the sub-classifiers operate on relatively small number of classes, the overall domain can be expanded by adding more sub-domains without compromising the accuracy.

3.2. Categorical Partition

The rules by which we split the domain need to be simple and lead to a feasible and accurate selection method in the first layer. This split is often natural in S2S applications as the data often belong in categorical groups. For example, data from the patient care domain [1] have categories like emergency, cardiology, neurology, etc. The data that we used in this work was collected for *DARPA's Transtac* project and consists of questions about people (biographic information, work, education), places (houses, land), civil affairs (water, sewage, electricity),and so on (Fig. 1). Therefore domains are naturally split across the category boarders. In addition, since frequent changes of category hardly happens throughout a discourse, an accurate mechanism (first layer) can be developed to capture and track the categorical association of the exchanged utterances. Occasional overlaps among categories can happen.

Category: Bio

Class 1:
What is his full name?
What is his name?
What is his complete name?
.....
Class 6:
What is his address?
Where does he live?
What street does he live on?
.....

Category: Electricity

Class 203:
How is your electricity?
How is your electricity these days?
How is your electricity working?
.....
Class 204:
Where do you get your electricity from?
What is the source of your electricity?
Where does your electricity come from?
.....

Figure 1: Sample of categorized training data for concept classification

3.3. Category Detection

As mentioned above, for the lower layer of classification (concept level), maximum likelihood classifiers can be employed as in (1). For the top layer we explore two alternative methods as follows.

3.3.1. Maximum Likelihood Category Classifier

The classifier of (1) can be trained to operate in the category level by using all the data from different classes in each category to build the LM for that category. Here, conceptual classes are replaced with categorical ones.

3.3.2. Category Detection by Topic Modeling

The semantic association of words within the categories of training data can be captured by topic modeling. If we consider the data in each sub-domain as a document, LDA can be applied to learn the topics and represent each sub-domain (document) by a probability distribution over these topics [5].

With LDA each word of a document is assumed to be associated with a topic. For each topic, words can be randomly picked with a multinomial distribution that is specific to that topic. For each document, the probability distribution of topics is also multinomial and specific to that document. The parameters of the latter distribution are considered to be samples of a Dirichlet random vector. Variational method [5] or Markov chain Monte Carlo (MCMC) [6] can be applied to extract the topics and associated distributions. Here we used the latter method both for training the models and to infer the topic distributions throughout a discourse.

Assume a dialog is in progress with a specific category, e.g., dermatology. Using a sufficient number of utterances we can infer the topic distribution $\mathbf{p}_d$ for that collection using the above method. By comparing this distribution to the distribution of each sub-domain, acquired from the training, we can assess the category of the discourse. The distributions can be compared using Kullback–Leibler divergence. If $\mathbf{p}_i$ represents the topic distribution of sub-domain i, the category of the discourse can be estimated as the category of sub-domain $\hat{g}$ where,

$$\hat{g} \triangleq \arg\max_{i=1\ldots K} \{D_{\mathrm{KL}}(\mathbf{p}_i \parallel \mathbf{p}_d)\} \tag{4}$$

and K is the number of categories.

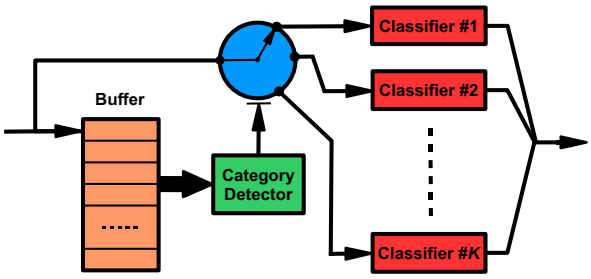

Figure 2: Two-layered classification scheme

With topic modeling in the first layer we capture the global semantic association of the words in a sub-domain or discourse. We use this information to find the category of the discourse and select the correct classifier exclusively trained for that category. The classifiers in the second layer, however, operate based on the local semantic and syntactic dependencies via class LMs.

3.4. Buffering

Without a highly accurate category detection, the two-layered structure will not be beneficial. The category of the dialog often remains constant over multiple rounds of utterance exchange. Therefore it is reasonable to use a few buffered utterances for that purpose rather than relying on a single one (Fig. 2).

For the first couple of conversational exchanges the category cannot be detected reliably therefore it is more appropriate to use the outcome of a single-layered classifier. It is customary in S2S applications to provide the user with a short list of hypothetical translations. In that case, for the first few sentences the results from both two-layered and single-layered classifiers could be presented to the user. As the discourse progresses and more information is gathered, the detection becomes more accurate.

4. Experiments and Results

4.1. Data

We used the data that were collected for phase one of DARPA's Transtac project. The data is in the form of 64,342 questions grouped as paraphrases in 621 classes and 38 categories, such as biographic information, education, family, finance, etc. Most of the categories include less than 2,000 questions. We remove 4 categories with more than 4,000 questions to avoid biasing toward them. The data from the remaining 34 categories consisted of 11,409 questions in 441 classes were used here. Fig. 1 shows a sample of that data set. From the English side of the Transtac parallel corpus, 654K sentences (5.5M words) were used to build the background LM for the classifiers.

4.2. Oracle Test

To confirm that a two-layered classification was potentially more accurate than a conventional one, we ran an oracle test in which always the correct category was chosen. For that purpose we randomly selected a test set of 2,000 questions and used the rest for training. One question per class was kept out of the selection process to guarantee that every class at least have one sentences for training.

The distribution of classes and sentences over categories are shown in Fig. 3 (a) and (b), respectively, for the training data. For the categories that were eliminated from the original data, the charts show zero number of classes and sentences. It is clear

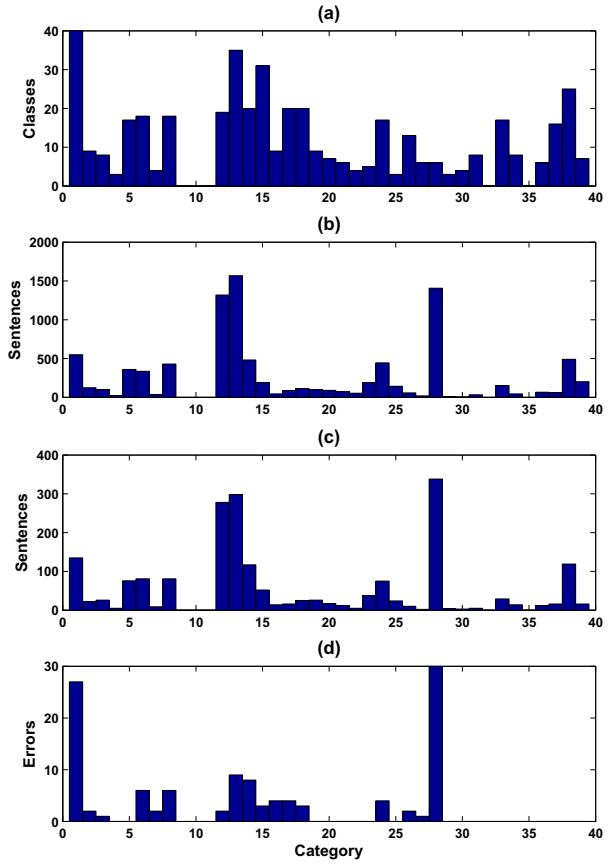

Figure 3: (a) Number of classes per category for training data, (b) Overall number of sentences per category for training data, (c) Overall number of sentences per category for testing set, (d) Number of errors per category for testing set

that more classes in a category is not necessarily accompanied by a higher number of training sentences for that category. As Fig. 3 (c) shows, the distribution of sentences in the test set closely resembles the training set which has been the purpose of the random selection.

The error rate for the conventional single-layered classifier was 7.6% while for the sub-domain classifiers with the oracle category selection was lower by 25%, relative (1.9% absolute). This significant error rate reduction shows the benefit of splitting the domain if an accurate category detection mechanism exists. The number of errors for each sub-classifiers is shown in Fig. 3 (d).

4.3. Topic Modeling of Categories

We used the same data sets of the previous experiment for topic modeling of the categories using the *Mallet* toolkit [9]. The Gibbs sampling [6] method is adopted in Mallet for LDA model training and inference. We used the training data to build the topic models which were then used to find the topic distribution of the test set. The detection was performed as explained by (4).

At first we set the topics number equal to 100 as it is a common choice in the literature [6], removed the stop words from the data, and shut off the procedure that optimizes the Dirichlet parameters. We ran 100,000 iterations of Gibbs sampling for training and 10,000 for inference. Using all the test set the detection process captured all the 34 categories correctly.

To investigate the effect of the topics number, we ran a set of experiments with different values for that parameter. In that

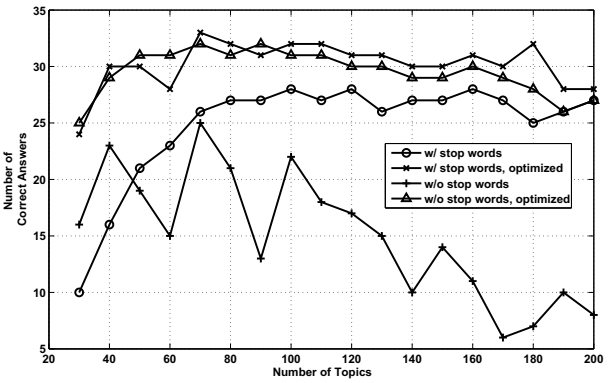

Figure 4: Category detection using topic modeling with different parameters

case we only used at most two sentences per category from the test set, which was equivalent of using a buffer with size two in Fig. 2. That set of experiments was also repeated with parameter optimization and/or leaving the stop words in the data. For each experiment, the number of iterations was the same as the above experiment. The optimization of Dirichlet parameters was carried out in every 500 iterations with an initial burn-in period of 1,000 for the cases that involved such a procedure.

The results are shown in Fig. 4. It is clear that having the stop words in the data has been quite advantageous. Also, optimization of Dirichlet parameters has significantly improved the performance. The best results were achieved for 70 topics with optimized parameters while leaving the stop words in.

4.4. Effect of Buffer Size on Category Detection

We examined the performance of both category detection methods for different buffer sizes. For testing, we selected five random sentences from each category and used the rest for training. To avoid having classes with no training data, one sentence per class was held out of the selection pool. The training data were used to build LMs for the categorical classifier as well as topic models for each category. For topic modeling we set the number of topics to 70 and ran the Gibbs sampling method with optimization for 100,000 iterations, without removing the stop words. For buffer sizes of one and two we repeated the test five times and twice, respectively, with different test utterances, and averaged the results.

Fig. 5 shows the number of correct decisions by two methods for different buffer sizes. Both methods detected all the categories correctly for buffer sizes of four and five. This indicates that the hierarchical classification method could reach the oracle results with a very small buffer size. With no buffering involved, the LM-based method had a better accuracy compared to detection with topic models (at the 0.05 significance level).

5. Conclusions

Scaling up the concept classifier is possible through a hierarchical structure without significantly degrading accuracy. Such structures can be built based on the categorical partitions of the domain. We showed the benefit of such partitioning and proposed to use two promising category detection methods in a two-layered classification scheme. These two methods—language model based classification and topic modeling—use the discourse information to select the correct concept classifier that operates on a sub-domain.

The results of the above two methods reveal cases of mismatch in their errors, therefore combining their results to benefit from both is a part of our ongoing work. We are planning to deploy the topical n-gram modeling method [10] to incorporate more ordering information in the process. We are also investigating the use of more sophisticated metrics such as Hellinger distance for a more precise comparison of topic distributions.

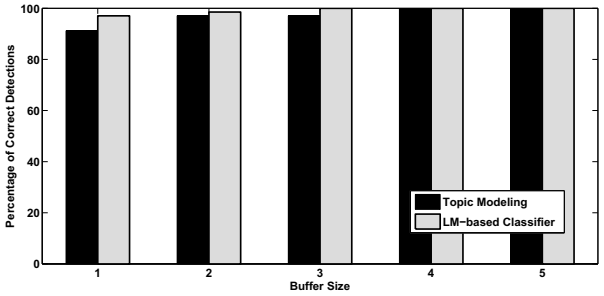

Figure 5: Effect of buffer size on the LM-based classification and Topic Modeling

6. Acknowledgments

This work was supported in part by funds from DARPA and NSF.

7. References

[1] E. Ettelaie, P. G. Georgiou, and S. Narayanan, "Mitigation of data sparsity in classifier-based translation," in *Proc. of Coling 2008 Workshop on Speech Processing for Safety Critical Translation and Pervasive Applications*, Manchester, UK, August 2008, pp. 1–4.

[2] F. Ehsani, J. Kinzey, D. Master, K. Sudre, D. Domingo, and H. Park, "S-MINDS 2-way speech-to-speech translation system," in *Proc. of the Medical Speech Translation Workshop, Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology (NAACL-HLT)*, New York, NY, USA, June 2006, pp. 44–45.

[3] A. Leuski, J. Pair, D. Traum, P. J. McNerney, P. Georgiou, and R. Patel, "How to talk to a hologram," in *Proc. of the Eleventh international conference on Intelligent user interfaces (IUI)*, Sydney, Australia, January-February 2006, pp. 360–362.

[4] E. Ettelaie, P. G. Georgiou, and S. Narayanan, "Cross-lingual dialog model for speech to speech translation," in *Proc. of the Ninth International Conference on Spoken Language Processing (ICSLP)*, Pittsburgh, PA, USA, September 2006, pp. 1173–1176.

[5] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *Journal of Machine Learning Research*, vol. 3, pp. 993–1022, March 2003.

[6] M. Steyvers and T. Griffiths, "Probabilistic topic models," in *Handbook of Latent Semantic Analysis*, T. Landauer, D. Mcnamara, S. Dennis, and W. Kintsch, Eds. Mahwah, NJ, USA: Lawrence Erlbaum Associates, Inc., 2007.

[7] S. Banerjee, "Improving text classification accuracy using topic modeling over an additional corpus," in *Proc. of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, Singapore, Singapore, July 2008, pp. 867–868.

[8] W. Sriurai, P. Meesad, and C. Haruechaiyasak, "Hierarchical web page classification based on a topic model and neighboring pages integration," *International Journal of Computer Science and Information Security (IJCSIS)*, vol. 7, no. 2, pp. 166–173, February 2010.

[9] A. K. McCallum, "Mallet: A machine learning for language toolkit," 2002, http://mallet.cs.umass.edu.

[10] X. Wang, A. McCallum, and X. Wei, "Topical N-Grams: phrase and topic discovery, with an application to information retrieval," in *Proc. of the Seventh IEEE International Conference on Data Mining (ICDM)*, Omaha, NE, USA, October 2007, pp. 697–702.