Towards Unsupervised Training of the Classifier-based Speech Translator

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

Signal Analysis and Interpretation Laboratory, Ming Hsieh Dept. of Electrical Engineering
Viterbi School of Engineering, University of Southern California
ettelaie@usc.edu

Abstract

Concept classification has been proven to be a useful translation method for speech-to-speech translation applications. However, preparing training data for classifier is a cumbersome task for human annotators. An unsupervised training method is introduced here that is based on utterance clustering. A technique to measure the distance between two utterances, based on the concepts they express, along with an appropriate clustering method has been adapted.

Index Terms: speech to speech translation, utterance clustering, concept classification

1. Introduction

Statistical machine translation (SMT) is the most commonly used translation technique for speech-to-speech translation systems [1, 2]. The statistical models that are deployed in these methods provide a great deal of flexibility that results in good coverage of the domain. However, translations are not always fluent. In addition, the source utterances are the output of a speech recognizer, and hence the noisy input intensifies the degradation of the quality of the translation output.

The main goal in speech to speech applications is to facilitate the accurate exchange of the concepts between the interlocutors rather than producing a word by word (literal) translation of the source. Existing SMT systems are typically optimized for faithful translation rather than this form of interpretation. A well defined dialog domain can be partly covered by a number of predefined concept classes. If each class (concept) is represented by a target language utterance, the translation task boils down to classifying the source utterances based on the concepts they convey [1, 3]. Therefore a classifier can be added as an interpreter by mapping the input utterance to one of the predefined concepts and transfer a previously generated fluent translation of this concept, which we will call the *canonical* cluster representation. This quantization of the concept space has obvious advantages and disadvantages. The major disadvantage is the inability to cover the complete space of all possible concepts. The advantages include the ability to deal with and likely correct noisy ASR output and the ability to generate significantly more fluent output, if operating within domain. The parallel combination of both concept classification – providing high quality in a small domain – and SMT – providing large coverage – can leverage the strength of both methods [1, 3]. This also requires rejection by the classifier when it detects out-of-domain concepts as presented in [3].

The first step to build a concept classifier is to select a set of concepts, which we call canonical cluster representations, and which can be thought of as the “quantization levels” of the domain. These canonical representations should cover a reasonable portion of the dialog domain. The second step is to group (or “quantize”) the remaining training data into this set.

The size of currently available data corpora is an incentive to seek automatic ways of both selecting the concept classes and automatic clustering of the training space into these classes. The reason that two sentences are placed in the same concept class motivates the definition of a distance metric between them. From a translation point of view, sentences share the same concept if they have “similar” translations. With the existence of an appropriate concept distance metric the problem of unsupervised training of the classifier would simply reduce to a clustering problem. The focus of this work is twofold: 1) identify a cross-sentence distance metric that will correlate well with the concept closeness of the two sentences in question, 2) identify and employ clustering techniques that rely on relative rather than global distance metrics.

The following section reviews the translation technique by means of concept classification, along with its training procedure. In section 3 the proposed method for unsupervised training is explained in detail. Section 4 covers the evaluation of the training procedure. Both intermediate and end-to-end evaluation measures are discussed. Section 5 consists of the experiment details and the associated data used in this work. Both the proposed method and the k-means clustering algorithm were investigated. The results are compared and discussed in section 6 which is followed by conclusion in Section 7.

2. Concept classifier

Concept classification has been successfully used in speech to speech translation systems as an alternative translation method in addition, and in parallel to, the more traditional statistical machine translation techniques. Also, it is often used in virtual interactive character systems implementing speech understanding in machine spoken dialog system, e.g., [4].

While the training of an SMT is mainly done through the use of bilingual parallel data, training for a concept classifier requires sets of same (or very similar) concept data. These are often generated by deciding a set of canonical utterances and then for each one of them manually generating a large number of paraphrases and significantly similar concepts. This procedure can be extremely time consuming.

One approach for classification using these training data is the bag-of-words approach, where each class generates a bag of words. Classification consists of finding which BOW best represents the input sentence.

A more mathematically rigorous approach is to build a *Language Model* (LM) for each concept class. The classifier selects the appropriate class based on the maximum likelihood criterion, as follow,

$$\hat{c} = \arg\max_{c \in C} \left\{ P_c \left(\mathbf{e} \mid c \right) \right\} \tag{1}$$

where C is the set of concept classes, $\mathbf{e}$ is the input utterance and $P_c(\mathbf{e} \mid c)$ is the score of $\mathbf{e}$ from the LM of class c.

This LM based method of classification was used in the Transonics system [1] which was developed as an English/Farsi speech translator in the doctor-patient interaction domain. For that system, the canonical concepts were manually selected using medical phrase books, websites, and by human judgment. Then human subjects were asked to provide paraphrases through a website, a web-based game, and paraphrasing sessions. The lessons learned can be summarized as,

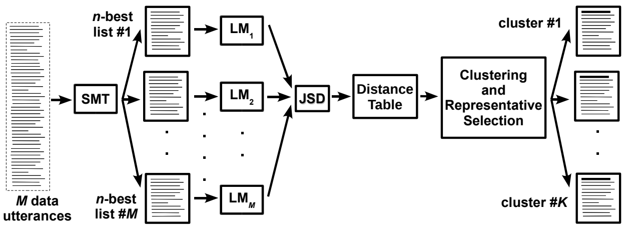

Figure 1: Overview of the proposed data preparation procedure

1. Manually selecting the concepts can lead to a poor coverage of the dialog domain.
2. Since the selected concepts are not driven from real data, some of them might be uncommon in real dialogs.
3. The overlapping concepts are very difficult to avoid.
4. For moderate number of classes, the training data are difficult to collect from human resources, and the method is not practical for large number of classes.
5. The paraphrases provided by human subjects are not always common sentences in the dialog domain.

However, the demonstrated suitability of classifier-based translation engines for the Transonics system is a strong incentive to seek new approaches to overcome some of the above obstacles.

3. Unsupervised training

The idea for unsupervised training of the classifier is to use a large amount of in-domain data, such as the ones that are already available for the training of the speech recognizer language models (monolingual), and if available, the bilingual data for SMT training. The goal is to identify the common concepts in these data sets, and for each concept, create a cluster containing all the utterances that convey that concept. Human input could be employed at the final step to represent the concepts in a canonical form in the target language. Obvious requirements for implementing this procedure are first specifying a distance metric among phrases and second, choosing a clustering method.

3.1. Utterance level distance – Handling Sparseness

Various methods of document comparison have been introduced and deployed in a wide range of applications. The essence of these methods is a measure that indicates the similarity of the documents. For instance, in text clustering, documents are represented by points in a vector space. For each document, a vector is generated in a fashion where words represent the dimensions and number of occurrences, the scales [5]. Then, distance measures, e.g., Euclidean distance, can be used as a similarity metric. The vectors, however, will contain no word ordering information.

In attempting to compare utterance distance however, the vector would be so sparse making the comparison practically impossible. For example say a medium sized domain contains N words where $N > 30,000$, but a sentence, represented by vector $\mathbf{x}$, will contain at most 10-20 words. As a result, direct comparison of the two N-dimensional vectors would unlikely provide any meaningful distance metric.

Here, the proposed solution is to "fuzzify" the vector representing the concept by adding a large number of similar but noisy measurements to it, $\mathbf{x}_i = \mathbf{x} + \mathbf{n}$; thus, the representation will become $\mathbf{x} + \mathbf{x}_1 + \ldots + \mathbf{x}_K$. The goal is to reduce the sparsity of the measurement while attempting to keep the noise (at a conceptual level) to a minimum.

One way to create $\mathbf{x}_i = \mathbf{x} + \mathbf{n}$ is by employing an SMT translation engine in a language pair with plentiful available resources, and in that case $\mathbf{x}_i$ would be a representation in another domain – namely the target language. In SMT methods, a combination of scores from different models are often used to rank hypothetical candidates and n-best lists could be generated accordingly [6]. The hypothesis here is that an SMT built on a vast amount of data would provide little degradation as one moves further down the n-best list, and hence $\mathbf{x}_1, \mathbf{x}_2, \ldots \mathbf{x}_K$ will contain acceptably low levels of noise.

3.2. Distance Measure

With the existence of a large number of noisy measurements, the problem of comparison can now be reformulated. The sparseness is significantly reduced and hence the utterance represented by $\mathbf{x}$ can now be represented by its fuzzified language model.

The classifier presented by Eq. 1 was aimed at comparing utterances based on class LM's. In this case we have LM-LM comparison, and since these models are approximations of probability density functions, they can be compared using information theoretic measures like relative entropy (*Kullback-Leibler divergence* – KLD). Although relative entropy is not commutative and therefore could not be used as a metric, it can be modified in the following way to serve the purpose.

$$KLD_{sym}(P, Q) = \frac{1}{2}D(P \parallel Q) + \frac{1}{2}D(Q \parallel P) \qquad (2)$$

Jensen-Shannon divergence (JSD) [7] is another symmetric and smoother derivation of the relative entropy. It is defined as,

$$JSD(P, Q) = \frac{1}{2}D(P \parallel M) + \frac{1}{2}D(Q \parallel M) \qquad (3)$$

where $M = \frac{1}{2}(P + Q)$. A recursive algorithm has been presented in [8] to efficiently calculate the relative entropy (and hence KLD and JSD) between language models.

3.3. Clustering Algorithms

Selecting concepts and forming classes of data that could be used for classifier training is in fact a clustering problem. While each cluster represents a concept class, the utterances forming these clusters can be used to train the classifier.

Most of the the partitional algorithms that are used for document clustering rely on vector space representation. For instance k-means algorithm and all its variations are based on centroid computations that are only meaningful when the items are represented by vectors. This is despite the variety of ways that the vectors might be defined and the variations in processing details such as normalization, inverse document frequency (idf) adjustment, etc.

Here, the goal is to use language models and an information theoretic measure among them as the similarity metric. In that case, only algorithms could be applied that rely on the distance among the items as the sole form of information, as the items coordinates are not defined. One of the algorithm with this feature is the Exchange method introduced in [9].

In the Exchange method the clustering problem is reformulated as an optimization task. If c_1, c_2,..., c_K are the clusters, the goal is to minimize the cost function below by moving the items around the clusters. For a cluster set $C \triangleq \{c_1, c_2, \ldots, c_K\}$,

$$\Phi(C) = \sum_{i=1}^{K} \frac{1}{|c_i|} \sum_{\substack{x, y \in c_i \\ x \neq y}} d(x, y) \qquad (4)$$

Here, $d(x, y)$ is the distance between items x and y, and $|\cdot|$ is the cluster cardinality. To avoid local minima, the Exchange method should be run several times with different random initializations.

3.4. Concept Representatives

Each data cluster is associated with a concept that needs to be represented by a canonical utterance. A human supervisor can generate this representative or manually draw one from the cluster members. When the number of clusters is too large, a method similar to the one used in [10] can be applied to select the representatives automatically. In each cluster, an utterance is selected that has the least accumulative distance with the other members of that cluster.

Table 1: The results of different clustering schemes

Method	Agreement [%]	1-Agreement [%]	Ave. Entropy [bits]	Acc. [%]	Acc. 4-best [%]
Random	97.5	2.5 (baseline)	3.785	14.3	33.9
Exchange method with KLD ($n = 2,000$)	98.4	1.6 (36%)	5.158	45.7	63.6
Exchange method with JSD ($n = 2,000$)	98.4	1.6 (36%)	5.115	47.8	61.7
Spherical k-means with original data	97.9	2.1 (16%)	4.763	38.3	52.1
Spherical k-means with n-best documents	98.1	1.9 (24%)	4.843	37.3	53.7
Reference annotation	100.0	0	6.213	69.4	86.0

3.5. Training

What was described in the above sub-sections, can be put together as an automatic method for concept selection and training data preparation for a concept classifier. The steps are illustrated in Fig. 1 and are implemented as follows,

1. *Domain Definition:* Utterances from the source language data are selected.
2. *Sparseness Reduction:* An SMT system is used to translate these utterances to the target or any other language.
3. *Statistical Representation:* Language models are built from n-best lists provided by the SMT system.
4. *Distance Metrics:* A table of JSD or KLD measures are built for all the possible language model pairs.
5. *Clustering:* Exchange method is applied using the above distance information to cluster the original utterances.
6. *Representative Selection:* For each cluster, a representative is chosen. The representative might be translated (manually or by an SMT) or pulled out from the target language part of data, in case of parallel corpus. If the classifier selects a certain class, the translated representative of that class would be the output of the overall system.

4. Evaluation

The classifier accuracy is the main evaluation measure, however, measuring the clustering quality, as an intermediate level assessment is beneficial for developing the training method.

4.1. Clustering Evaluation

Two methods for evaluating the quality of the clustering task have been used in this work. The first one is introduced in [11] and is based on computing the percentage of binary decisions that are common between the clustered and reference data: every possible pair of items in the data, gives a correct/wrong output depending on the agreement of the reference with the hypothesis regarding to their same-cluster status. A percentage of the decisions that are in agreement with the reference data can be used as an indicator of the quality of a clustering task. Note that this measure considers placing two items from different classes in two different clusters, a correct decision, and hence is highly biased toward the correct measurements. For example, with 100 classes, a decision about two given items from different classes, has 99% chance of being right. Hence, this measure is not expected to distinguish performance improvements.

The second evaluation method is based on measuring the cluster purity, i.e., the average entropy of clusters. If R is the set of reference classes, the average entropy is defined for cluster set C as,

$$E = -\sum_{c\in C} \frac{|c|}{|C|} \sum_{r\in R} P_{cr} \log(P_{cr}) \qquad (5)$$

where,

$$P_{cr} \triangleq \frac{|c \cap r|}{\left| c \cap (\bigcup_{\rho\in R} \rho) \right|} \qquad (6)$$

4.2. Overall evaluation

After training the classifier with the clustered data, an annotated test set can be used to measure the success level of the automatic training. To measure the classification accuracy, it suffices to count the number of cases that input utterance and the classifier output are from the same class in the reference annotations.

In the speech-to-speech translation systems it is common to provide the user with multiple options. For instance in the system of [1], the user is given a list of 4-best classifier outputs to choose from. Therefore, it is also useful to measure the classifier accuracy within its n-best outputs. However, to avoid the accuracy bias, when that n-best list contains multiple correct answers, only one of them is counted.

5. Data and Experiments

The proposed method was examined through a set of experiments and its effectiveness was compared with the supervised training.

The available data set was originally collected and used in the Transonics project [1]. That data set was formed in the following way: First the concepts were carefully chosen using experts' judgments and medical handbooks. Then the associated data set for each concept was collected using a website, a web-based game, and several paraphrasing sessions at the Information Sciences Institute of the University of Southern California.

Although there were 1,269 classes available in the English side, the system was evaluated by using 97 most richly paraphrased ones. The associated 1,207 source (English) utterances were then randomly split into 500 for training and 707 for testing.

For the generation of the target language n-best lists we employed a phrase-based SMT [12] trained on a general domain English/Farsi parallel corpus with 1.2M English words. To study the effect of the length of the n-best list, n-best lists with 500, 1000, 2000, and 3000 hypotheses per source sentence were generated.

The language models were generated using the SRILM toolkit [13]. The KLD and JSD distance tables were formed by applying the algorithm from [8] to every pair of such language models. The exchange method was used to cluster the utterances. Throughout this work the number of classes was always considered a known parameter. Investigating the methods to estimate that parameter is part of the succeeding work.

As the baseline for clustering process, the spherical k-means algorithm was also employed using gmeans software [14]. For that purpose, the MC toolkit [15] was first used to create the vector models from the documents (n-best lists). In addition, random clustering, i.e. the input utterances randomly and uniformly dispersed over the 97 output clusters, was also used for comparison.

Table 1 consists of the results of experiments with different clustering methods. The Exchange method with both KLD and JSD metrics was tried. For these experiments the length of the SMT n-best list was set to 2,000. The results from spherical k-means clustering algorithm are also included in Table 1. For comparison, k-means was applied to both the original English utterances and their associated n-best list documents. Table 1 also contains the results from both the supervised training using data annotations and a random clustering scheme. In each case the clustering agreement and average entropy were measured.

Since the main goal was to build a classifier based translator, following each of the above cases, a classifier was trained based on the resulting clusters and its accuracy was measured using the testing data. Along with the classifier accuracy, the accuracy within its 4-best outputs was also measured in each case (Table 1).

Table 2: The effect of the length of the SMT n-best list using Exchange method with JSD metric

n-best length	500	1,000	2,000	3,000
Agreement [%]	98.3	98.3	98.4	98.3
Ave. Entropy [bits]	5.060	5.091	5.115	5.088
Accuracy [%]	44.1	45.0	47.8	46.0
Accuracy within 4-best [%]	61.2	60.4	61.7	60.1

In Table 2 the results of experiments with different lengths of n-best lists are reported. In this set of experiments the Exchange method was carried out with JSD metric and the clustering agreement and average entropy were measured for each case. Table 2 also shows the classifier accuracy and its accuracy within 4-best outputs measured on the testing data.

6. Results and Discussion

The results in Table 1 provide information that can help the further development of an unsupervised training method. The fifth column clearly shows that the proposed method, which was motivated by the needs of the translation task, led to a higher classification accuracy compared to the spherical k-means method (absolute 10.5% improvement). It also shows that the attempt to apply the k-means algorithm on the n-best lists did not produce any better results than the original k-means algorithm. The gap between the results from supervised and unsupervised training is also prominent (69.4% vs. 47.8% for Exchange method with JSD).

The clustering evaluation measures are shown in the columns two to four of Table 1. It is obvious that the agreement measure has not been useful due to its rapid saturation, although $1-$Agreement can be a useful relative metric as shown in the third column. On the other hand the variation in average entropy follows the classification accuracy trend (column four). Therefore it seems to be useful as an intermediate evaluation measure for future developments.

The results also indicate that the KLD metric produced around 2% higher accuracy in 4-best classifier outputs, compared to the JSD metric. Therefore the former metric is a better choice for applications that provide the user with multiple options because it increases the chance of finding the right answer among them.

The utterances are not evenly distributed over the classes as some of the concepts (greetings, etc.) are much more frequent than the others. This feature is preserved in the training and the testing sets and leaves some classes with more items. Even in a random clustering scheme, these items dominate some of the clusters. While testing, they help a correct classification of the items from the frequent classes and maintain a 14.3% accuracy (random clustering).

The effect of the n-best list length on the classifier accuracy is shown in Table 2. The clustering quality (and hence the classification accuracy) increased as the n-best list length grew from 500 to 2,000 but decreased for the length of 3,000. As more SMT hypotheses are included in the clustering process a better clustering is achieved due to the lexical diversity. However, at some point the assumption that all the hypotheses are quality translations of the source utterance, loses its validity. Low quality hypotheses in the bottom of the list cause an inferior clustering result. Table 2 also indicates the same trend, with less intensity, for the accuracy within 4-best outputs as well as the clustering evaluation measures.

7. Conclusion

Using the proposed method, the concept classifier can be trained automatically. The price is of course the significantly lower accuracy compared to what could be achieved by supervised training.

This work is the first step in the development of an unsupervised training methods for classifier based translation systems. Semiautomatic assessment of the class numbers and using filtering in different stages (original data, n-best lists, clusters, etc.) to improve the training process are part of the research that is currently in progress.

8. Acknowledgment

The authors thank Tom Murray for his help with gmeans software. Research was support in part with funds from DARPA.

9. References

[1] S. Narayanan, S. Ananthakrishnan, R. Belvin, E. Ettelaie, S. Ganjavi, P. Georgiou, C. Hein, S. Kadambe, K. Knight, D. Marcu, H. Neely, N. Srinivasamurthy, D. Traum, and D. Wang, "Transonics: A speech to speech system for english-persian interactions," in *Proc. of IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, St.Thomas, U.S. Virgin Islands, November-Decmeber 2003, pp. 670–675.

[2] R. Hsiao, A. Venugopal, T. Kohler, Y. Zhang, P. Charoenpornsawat, A. Zollmann, S. Vogel, A. W. Black, T. Schultz, and A. Waibel, "Optimizing components for handheld two-way speech translation for an english-iraqi arabic system," in *Proc. of the Ninth International Conference on Spoken Language Processing (ICLSP)*, Pittsburgh, PA, USA, September 2006, pp. 765–768.

[3] 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 (ICLSP)*, Pittsburgh, PA, USA, September 2006, pp. 1173–1176.

[4] D. Traum, A. Roque, A. Leuski, P. Georgiou, J. Gerten, B. Martinovski, S. Narayanan, S. Robinson, and A. Vaswani, "Hassan: A virtual human for tactical questioning," in *Proc. of the 8th SIGdial workshop on Discourse and Dialogue*, Antwerp, Belgium, September 2007, pp. 75–78.

[5] I. S. Dhillon and D. S. Modha, "Concept decompositions for large sparse text data using clustering," *Machine Learning*, vol. 42, no. 1, pp. 143–175, Jan 2001.

[6] H. Ney, S. Nießen, F. J. Och, C. Tillmann, H. Sawaf, and S. Vogel, "Algorithms for statistical translation of spoken language," *IEEE Trans. on Speech and Audio Processing, Special Issue on Language Modeling and Dialogue Systems*, vol. 8, no. 1, pp. 24–36, January 2000.

[7] J. Lin, "Divergence measures based on the shannon entropy," *IEEE Transactions on Information Theory*, vol. 37, no. 1, pp. 145–151, January 1991.

[8] A. Sethy, S. Narayanan, and B. Ramabhadran, "Measuring convergence in language model estimation using relative entropy," in *Proc. of the Eight International Conference on Spoken Language Processing (ICLSP)*, Jeju Island, Korea, October 2004, pp. 1057–1060.

[9] H. Sp̈ath, *The Cluster Dissection and Analysis Theory FORTRAN Programs Examples*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1985.

[10] H. Ye and S. Young, "A clustering approach to semantic decoding," in *Proc. of the Ninth International Conference on Spoken Language Processing (ICLSP)*, Pittsburgh, PA, USA, September 2006, pp. 5–8.

[11] K. Wagstaff and C. Cardie, "Clustering with instance-level constraints," in *Proc. of the Seventeenth International Conference on Machine Learning*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., June-July 2000, pp. 1103–1110.

[12] P. Koehn, F. Och, and D. Marcu, "Statistical phrase-based translation," in *Proc. of the Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology (NAACL-HLT)*, vol. 1, Edmonton, AB, Canada, May-June 2003, pp. 48–54.

[13] A. Stolcke, "Srilm - an extensible language modeling toolkit," in *Proc. of the International Conference on Spoken Language Processing (ICSLP)*, Denver, CO, USA, September 2002, pp. 901–904.

[14] I. S. Dhillon, Y. Guan, and J. Kogan, "Iterative clustering of high dimensional text data augmented by local search," in *Proc. of the IEEE International Conference on Data Mining (ICDM)*, Maebashi City, Japan, 2002, pp. 131–138.

[15] I. S. Dhillon, J. Fan, and Y. Guan, "Efficient clustering of very large document collections," in *Data Mining for Scientific and Engineering Applications*, V. K. R. Grossman, C. Kamath and R. Namburu, Eds. Kluwer Academic Publishers, 2001, pp. 357–381.