The UvA System Description for IWSLT 2010

Spyros Martzoukos, Christof Monz

ISLA, University of Amsterdam
Science Park 904, 1098 XH Amsterdam
{s.martzoukos, c.monz}@uva.nl

Abstract

We describe the machine translation system of the University of Amsterdam, that was used to decode the Chinese→English test sets of the DIALOG task. It consists of typical phrase-based translation, SRILM 5-gram language, lexicalized and distance-based distortion and word penalty models which are manipulated according to a model adaption technique, based on the identification of subdomains of the provided data sets.

1. Introduction

We attempt to improve translation quality by identifying the subdomains for the provided data sets of the DIALOG task of IWSLT '10, which have a structure that is amenable to techniques often encountered in Statistical Machine Translation (SMT), namely model adaptation (see e.g. [1-7]). These data sets can be easily decomposed into subsets by splitting them into as many documents as there are dialogs. In principle, by performing clustering on the resulting collection, we can treat each cluster individually and apply model adaptation methods to enhance translation quality. In Section 2 we briefly mention some related work and outline the approach we developed for this task in Sections 3 and 4 and suggest possible improvements in Section 5.

2. Related Work

Model adaptation is a research area in SMT that deals with the retrieval and/or generation and exploitation of data, for the purpose of biasing the machine translation models towards the statistics that would favour translations closer to a certain context (*domain*). Such data can be either monolingual or parallel, be of the same (*in-domain*) or even different (*out-of-domain*) contexts and are generally used to modify the existing language and translation models. The adaptation strategies depend on the type and amount of data available and we briefly mention some here.

In [1] and [2] monolingual in-domain data were used to enhance translation performance by self-training techniques, i.e. by using the translation system's own output. Information retrieval methods were used in [3] by selecting out-of-domain parallel data and in [4] by cross-lingually finding in-domain target side monolingual data. In [5] it was shown that using separate translation models together with multiple language models outperforms translations generated by simply training from different domains. In [6] they found that interpolated language models give similar performance to the use of multiple language models, with the former being more efficient.

3. Our Approach

Unlike the strategies mentioned in the previous section, where additional data is used to augment the translation and language models, we concentrate on the provided data only, by identifying its subdomains and exploit the characteristics of each one individually. We focus on the DIALOG task, where we consider each dialog of the training, development and test sets as a *document*, the collection of which is then clustered. Each generated cluster is treated as a separate data set from which its domain-specific translation and language models are extracted. A test document/dialog is translated with respect to the models of the cluster it belongs to.

In particular, we split each of the training and development source sets and test sets into as many documents as there are dialogs. This collection of documents is clustered based on the vocabulary of the collection of source documents, resulting in K document clusters, with each document appearing in at most one cluster. The process of determining the value of K and the details of the clustering procedure we employed are explained below, and we proceed with outlining the subsequent steps of our approach. The training and development target sets are also split into documents in the same manner and each of these documents is placed in one of the K clusters: a target document is placed in cluster c if and only if its translation (source document) belongs to cluster c. In other words, a typical cluster contains pairs of training and development documents (i.e. source and target documents, which are translations of each other, of both training and development type), as well as test documents, all of which belong to the same domain. The aim is to exploit the common characteristics that data of the same domain share, and we thus proceed with treating each cluster separately. The concatenation of the training documents of each side (source/target) of a cluster, say c, forms the bitext for this cluster, from which a cluster-specific phrase-based translation model is generated. The target side of this bitext is used to create the cluster-specific language model for c. Similarly, the concatenation of the development documents of c is used to tune the weights of the corresponding feature functions of these cluster-specific models. Additionally for tuning, we use both baseline translation and language models, as well as the baseline distortion model, and their corresponding weights, including the word penalty, are carried over as initial values for this process (see Section 4 for baseline details). Finally the test documents of c are decoded with respect to the models and weights generated for this cluster.

We note that certain clusters may contain only pairs of training and/or development documents and since no decoding of test documents takes place, such clusters are insignificant to our approach. On the other hand, if a cluster contains test documents and pairs of training but no development documents, we then use the baseline development set to tune the weights of the cluster-specific models. The remaining case occurs when a cluster contains test documents but no pairs of training documents. In that case, we back off to the baseline models and decode the test documents with respect to the weights generated for this cluster, or decode the test documents with respect to the baseline translation system, if the cluster does not contain any pairs of development documents. Nonetheless, all clusters in our experiments did contain pairs of training documents.

The clustering process we have employed is based on information-theoretic concepts. Data compression and information theory are linked via rate distortion theory [12] and in [13] a principled approach to the issue of the selection of the 'right' distance measure was proposed. The latter gives rise to our clustering process which is carried out in two steps, as suggested in [10]. First, a divisive hierarchical algorithm [11] is employed to cluster the *vocabulary* of the documents, and based on these word clusters, an agglomerative hierarchical greedy algorithm [10] for 'hard' clustering (i.e. every element belongs to exactly one cluster) is then used to cluster the documents. Both algorithms use the joint probability distribution of a document and a word as their input and interpret the generalised Jensen-Shannon divergence as the 'distance' between clusters.

In particular let D, W be a collection of documents and its vocabulary respectively. The joint probability distribution of a document $d \in D$ and a word $w \in W$ is given by

$$p(d, w) = p(w|d)p(d) = \frac{1 + n(w, d)}{|W| + |d|} \cdot \frac{1}{|D|},$$

where $n(w, d)$ is the frequency of word w in document d, $|W|$ is the size of the vocabulary and $|d|$ is the number of words in d. We assume that each document in the collection is equiprobable and Laplace's rule of succession is used for smoothing the conditional probability $p(w|d)$. The quantity of interest is the mutual information between D and W, $I(D, W)$, which is the reduction in entropy of one variable knowing the other, and is defined by

$$I(D, W) = \sum_{d \in D} \sum_{w \in W} p(d, w) \log \frac{p(d, w)}{p(d)p(w)}.$$

We first cluster the vocabulary W, so that the obtained word clusters, $\tilde{W}$, satisfy

$$I(D, W) \approx I(D, \tilde{W}). \qquad (1)$$

The resulting joint distribution of D and $\tilde{W}$ is used to cluster the documents D, so that the obtained document clusters, $\tilde{D}$, satisfy

$$I(D, \tilde{W}) \approx I(\tilde{D}, \tilde{W}). \qquad (2)$$

Relations (1) and (2) can be obtained by repeated application of the agglomerative algorithm, but we chose to cluster the vocabulary with the divisive algorithm because it performs better on this task [11]. It is important to note that at every step of the divisive algorithm all clusters are re-computed, whereas for the agglomerative algorithm only one merging of a cluster pair takes place. Nonetheless, both algorithms minimize the same distance when re-computing/merging the clusters. For both cases it can be shown ([10], [11]) that mutual information is *lost* after each step. In other words the difference

$$\delta \equiv I(X, \tilde{Y}_{before}) - I(X, \tilde{Y}_{after}), \qquad (3)$$

where $I(X, \tilde{Y}_{before})$ and $I(X, \tilde{Y}_{after})$ are the information values before and after the re-computation/merging, respectively, is positive and is, in fact, equal to the generalised Jensen-Shannon divergence. Thus, the choice of the members of $\tilde{Y}_{after}$ when recomputing/merging clusters at each step should be such that δ is minimized.

For the divisive algorithm the number of clusters should be chosen by the user and for the agglomerative algorithm, if δ becomes relatively high at step n, the process should be stopped and the K resulting clusters at step n is the output of the algorithm.

4. Experiments

All our models are built using the open source toolkit Moses [14]. Our baseline model consists of typical phrase-based, language, distortion and word penalty models generated from the training set of the DIALOG Chinese→English task. In particular, the phrase-based model consists of bidirectional phrase translation and lexical weighting features as well as a phrase penalty feature. A 5-gram language model is built using the open source SRILM toolkit [8] and employs Kneser-Ney smoothing. The distortion model consists of a distance-based reordering feature and bidirectional, oriented lexical reordering features, conditioned on both source and target phrases. The corresponding weights of these features and the word penalty feature, are tuned with minimum error rate training [9]. The decoder employs a multi-stack architecture of size 100, and uses a beam to manage the search space. The maximum number of translation table entries per input phrase is set to 20 and the distortion limit to 6.

The training, development, ’09 test and ’10 test Chinese sets are all split into as many documents as there are dialogs. The resulting collection contains 398 training, 10 development, 27 ’09 test and 37 ’10 test documents with an average of 17 sentences per document. This collection is then clustered and the corresponding English clusters are constructed.

In Table 1 we report BLEU scores for Chinese → English (CRR) for both ’09 and ’10 test sets, where a small improvement is observed for the latter over the baseline system.

In our experiments we found the optimal size of word clusters to be around 40, resulting in 17 document clusters. The smallest cluster contains 3 documents (all of training type), and the largest one contains 68 documents (41 training, 8 development, 11 ’09 test and 8 ’10 test). Following [11], no pruning of words has been done, meaning that all words have been used to extract word clusters $\tilde{W}$ form the vocabulary W.

Chinese→English (CRR)	Baseline	Clusters
Test ’09	**0.2261**	0.2178
Test ’10	0.1603	**0.1629**

Table 1: *BLEU scores for the baseline and cluster-based systems on the ’09 and ’10 test sets of Chinese→English (CRR), ‘case+punc’.*

5. Conclusions

We attempted to improve translation quality by identifying the subdomains for the provided data sets of the DIALOG task. The homogeneity and the size of the data sets, particularly of the development set, suggest that this method may not be suitable for model adaptation. Nonetheless, the clustering process has been treated as a black box, and we intend to exploit the said information-theoretic algorithms more explicitly in our models; every component of these algorithms is equipped with a probability and by clustering n-grams instead of just unigrams, we could re-estimate/adapt the relevant quantities of the cluster-specific models.

6. Acknowledgements

This work has been funded in part by the European Commission through the CoSyne project FP7-ICT-4-248531 and the GALATEAS project CIP-ICT PSP-2009-3-250430.

7. References

[1] N. Bertoldi and M. Federico, “Domain adaptation for statistical machine translation with monolingual resources”, Proceedings of the Fourth Workshop on Statistical Machine Translation, pp. 182-189, 2009.

[2] N. Ueffing, G. Haffari, and A. Sarkar, “Transductive learning for statistical machine translation”, Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, pp 25-32, 2007.

[3] A.S. Hildebrand, M. Eck, S. Vogel, and A. Waibel, “Adaptation of the translation model for statistical machine translation based on information retrieval”. Proceedings of EAMT, pp. 133-142, 2005.

[4] M. Snover, B. Dorr, and R. Schwartz, “Language and Translation Model Adaptation using Comparable Corpora”, Proceedings of the Conference on Empirical Methods in Natural Language Processing, pp. 857-866, 2008.

[5] P. Koehn and J. Schroeder, “Experiments in domain adaptation for statistical machine translation” Proceedings of the Second Workshop on Statistical Machine Translation, pp. 224-227, 2007.

[6] H. Schwenk and P. Koehn, “Large and diverse language models for statistical machine translation”, Proceedings of The Third International Joint Conference on Natural Language Processing, pp. 661-666, 2008.

[7] G. Foster and R. Kuhn, “Mixture-model adaptation for SMT”, Proceedings of the Second Workshop on Statistical Machine Translation, pp. 128-135, 2007.

[8] A. Stolcke, “SRILM an extensible language modeling toolkit”, Proceedings of the Int. Conf. Spoken Language Processing, 2002.

[9] F.J. Och, “Minimum Error Rate Training in Statistical Machine Translation”, Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics, pp. 160-167 2003.

[10] N. Slonim and N. Tishby, “Document Clustering using Word Clusters via the Information Bottleneck Method”, Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval, pp. 208-215, 2000.

[11] I. Dhillon, S. Mallela and R. Kumar, “A Divisive Information-Theoretic Feature Clustering Algorithm for Text Classification”, Journal of Machine Learning Research 3, pp. 1265-1287, 2003.

[12] T.M. Cover and J.A. Thomas, “Elements of Information Theory”, John Wiley &amp; Sons, New York, USA, 1991.

[13] N. Tishby, F. Pereira, W. Bialek, “The information bottleneck method”, Proceedings of the 37-th Annual Allerton Conference on Communication, Control and Computing, pp. 368-377, 1999.

[14] P. Koehn, H. Hoang, A. Birch, C. Callison-Burch, M. Federico, N. Bertoldi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E.

Herbst, “Moses: Open source toolkit for statistical machine translation”, Proceedings of ACL Demo Session, pp. 177180, 2007.