Domain Adaptation to Summarize Human Conversations

Oana Sandu, Giuseppe Carenini, Gabriel Murray, and Raymond Ng
University of British Columbia
Vancouver, Canada
{oanas,carenini,gabrielm,rng}@cs.ubc.ca

Abstract

We are interested in improving the summarization of conversations by using domain adaptation. Since very few email corpora have been annotated for summarization purposes, we attempt to leverage the labeled data available in the multiparty meetings domain for the summarization of email threads. In this paper, we compare several approaches to supervised domain adaptation using out-of-domain labeled data, and also try to use unlabeled data in the target domain through semi-supervised domain adaptation. From the results of our experiments, we conclude that with some in-domain labeled data, training in-domain with no adaptation is most effective, but that when there is no labeled in-domain data, domain adaptation algorithms such as structural correspondence learning can improve summarization.

1 Introduction

On a given day, many people engage in conversations via several modalities, including face-to-face speech, telephone, email, SMS, chat, and blogs. Being able to produce automatic summaries of multi-party conversations occurring in one or several of these modalities would enable the parties involved to keep track of and make sense of this diverse data. However, summarizing spoken dialogue is more challenging than summarizing written monologues such as books and articles, as speech tends to be more fragmented and disfluent.

We are interested in using both fully and semi-supervised techniques to produce extractive summaries for conversations, where each sentence of a text is labeled with its informativeness, and a subset of sentences are concatenated into an extractive summary of the text. In previous work (Murray and Carenini, 2008), it has been shown that conversations in different modalities can be effectively characterized by a set of “conversational” features that are useful in detecting informativeness for the task of extractive summarization. However, because of privacy concerns, annotated corpora are rarely publicly available for conversational data, including for the email domain. One promising solution to this problem is domain adaptation, which aims to use labeled data in a well-studied source domain and a limited amount of labeled data from a different target domain to train a model that performs well in that target domain. In this work, we investigate using domain adaptation that leverages labeled data in the domain of meetings along with labeled and unlabeled email data for summarizing email threads. We evaluate several domain adaptation algorithms, using both a small set of conversational features and a large set of simple lexical features to determine what settings will yield the best results for summarizing email conversations. In our experiments, we do not get a significant improvement from using out-of-domain data in addition to in-domain data in supervised domain adaptation, though in the setting where only unlabeled in-domain data is available, we gain from using it through structural correspondence learning. We also observe that conversational features are more useful in supervised methods, whereas lexical features are better leveraged in semi-supervised adaptation.

The next section surveys past research in domain adaptation and in summarizing conversational data. In section 3 we present the corpora and feature sets we used, and we describe our experimental setting in section 4. We then compare the performance of different methods in section 5 and draw conclusions in section 6.

Proceedings of the 2010 Workshop on Domain Adaptation for Natural Language Processing, ACL 2010, pages 16–22,
Uppsala, Sweden, 15 July 2010. ©2010 Association for Computational Linguistics

2 Related Work

We give an overview first of work on supervised and semi-supervised domain adaptation, then of research on summarization of conversations.

2.1 Supervised Domain Adaptation

Many domain adaptation methods have been proposed for the supervised case, where a small amount of labeled data in the target domain is used along with a larger amount of labeled source data. Two baseline approaches are to train only on the source data or only on target training data. One way of using information from both domains is merging the source and target labeled data sets and training a model on the combination. A method inspired by boosting is to take a linear combination of the predictions of two classifiers, one trained on the source and one trained on the target training data. Another simple method is to train a predictor on the source data, run it on the target data, and then use its predictions on each instance as additional features for a target-trained model. This was first introduced by Florian et al. (2004), who applied it to multilingual named entity recognition.

The prior method of domain adaptation by Chelba and Acero (2006) involves using the source data to find optimal parameter values of a maximum entropy model on that data, and then setting these as a prior on the values of a model trained on the target data. They find improvement in a capitalizer that adapts using out-of-domain and a small amount of in-domain data versus only training on out-of-domain WSJ data. Similar to the prior method, Daume's MEGA model also trains a MEMM. It achieves domain adaptation through hyperparameters that indicate whether an instance is generated by a source, target, or general distribution, and finds the optimal values of the parameters through conditional EM (Daume and Marcu, 2006). A simpler method of domain adaptation, that achieves a performance similar to prior and MEGA, was proposed by Daume (2007) and successfully applied to a variety of NPL sequence labeling problems, such as named entity recognition, shallow parsing, and part-of-speech (POS) tagging. Furthermore, this approach is straightforward to apply by copying feature values so there is a source version, a target version, and a general version of the feature, and was found to be faster to train than MEGA and prior. For all these reasons, we use Daume's method and not the other two in our experiments.

2.2 Semi-supervised Domain Adaptation

Because unlabeled data is usually much easier to collect than labeled data in a new domain, semi-supervised domain adaptation methods that exploit unlabeled data are potentially very useful.

In self-training, a training set is used that is originally composed of labeled data, and repeatedly augmented with the highest confidence predictions on unlabeled data. McClosky et al. (2006) apply this in a domain adaptation setting for parsing: with only unlabeled data in the target Brown domain, and labeled and unlabeled datasets in the news domain (WSJ and NANC respectively), a self-trained reranking parser performs almost as well as a parser trained only on Brown labeled data. However, McClosky concludes that self-training alone is not beneficial, and most of the improvements they get over previous work on domain adaptation for parsing are due to using the reranker to select the candidate instances produced in each iteration of self-training. Thus, one of the issues addressed in this paper is to asses whether self-training is useful for domain adaptation.

A more sophisticated semi-supervised domain adaptation method is structural correspondence learning (SCL). SCL uses unlabeled data to determine correspondences between features in the two domains by correlating them with so-called pivot features, which are features exhibiting similar behaviors in the source and target domains. Blitzer applied this algorithm successfully to POS tagging (Blitzer et al., 2006) and sentiment classification (Blitzer et al., 2007). SCL seems promising for other tasks as well, for example parse disambiguation (Plank, 2009).

2.3 Summarization

We would like to use domain adaptation to aid in summarizing multi-party conversations hailing from different modalities. This contrasts with much of previous work on summarization of conversations, which has focused on domain-specific features (e.g., Rambow et al, 2004). We will treat summarization as a supervised binary classification problem where the sentences of a conversation are rated by their informativeness and a subset is selected to form an extractive summary. Research in meeting summarization relevant to our task has investigated the utility of employing a large feature set including prosodic information, speaker status, lexical and structural discourse features (Murray et al., 2006; Galley, 2006). For email summarization, we view an email thread as a conversation. For summarizing email threads, Rambow (2004) used lexical features such as tf.idf, features that considered the thread to be a sequence of turns, and email-specific features such as number of recipients and the subject line. Asynchronous multi-party conversations were successfully represented for summarization through a small number of conversational features by Murray and Carenini (2008). This paved the way to cross-domain conversation summarization by representing both email threads and meetings with a set of common conversational features. The work we present here investigates using data from both emails and meetings in summarizing emails, and compares using conversational versus lexical features.

3 Summarization setting

Because the meetings domain has a large corpus, AMI, annotated for summarization, we will use it as the source domain for adaptation and the email domain as the target, with data from the Enron corpus as unlabeled email data, and the BC3 corpus as test data.

3.1 Datasets

The AMI meeting corpus: We use the *scenario* portion of the AMI corpus (Carletta et al., 2005), for which groups of four participants take part in a series of four meetings and play roles within a fictitious company. While the scenario given to them is artificial, the speech and the actions are completely spontaneous and natural. The dataset contains approximately 115000 dialogue act (DA) segments. For the annotation, annotators wrote abstract summaries of each meeting and extracted transcript DA segments that best conveyed or supported the information in the abstracts. A many-to-many mapping between transcript DAs and sentences from the human abstract was obtained for each annotator, with three annotators assigned to each meeting. We consider a dialogue act to be a positive example if it is linked to a given human summary, and a negative example otherwise. Approximately 13% of the total DAs are ultimately labeled as positive.

The BC3 email corpus[1]**:** composed of 40 email threads from the World Wide Web Consortium (W3C) mailing list which feature a variety of topics such as web accessibility and planning face-to-face meetings. Each thread is annotated similarly to the AMI corpus, with three annotators authoring abstracts and linking email thread sentences to the abstract sentences.

The Enron email corpus[2]**:** a collection of emails released as part of the investigation into the Enron corporation, it has become a popular corpus for NLP research due to being realistic, naturally-occurring data from a corporate environment. We use 39 threads from this corpus to supplement the BC3 email data.

3.2 Features Used

We consider two sets of features for each sentence: a small set of conversational structure features, and a large set of lexical features.

Conversational features: We extract 24 conversational features from both the email and meetings domain, and which consider both emails and meetings to be conversations comprised of turns between multiple participants. For an email thread, a turn consists of a single email fragment in the exchange. Similarly, for meetings, a turn is a sequence of dialogue acts by the same speaker. The conversational features, which are described in detail in (Murray and Carenini, 2008), include sentence length, sentence position in the conversation and in the current turn, pause-style features, lexical cohesion, centroid scores, and features that measure how terms cluster between conversation participants and conversation turns.

Lexical features: We derive an extensive set of lexical features, originally proposed in (Murray et al., 2010) from the AMI and BC3 datasets, and then compute their occurrence in the Enron corpus. After throwing out features that occur less than five times, we end up with approximately 200,000 features. The features derived are: character trigrams, word bigrams, POS tag bigrams, word pairs, POS pairs, and varying instantiation ngram (VIN) features. For word pairs, we extract the ordered pairs of words that occur in the same sentence, and similarly for POS pairs. To derive VIN features, we take each word bigram w_1,w_2 and further represent it as two patterns p_1,w_2 and w_1,p_2 each consisting of a word and a POS tag.

3.3 Classifier

In all of our experiments, we train logistic regression classifiers using the *liblinear* toolkit[3]. This choice was partly motivated by our earlier summarization research, where logistic regression classifiers were compared alongside support

[1] http://www.cs.ubc.ca/labs/lci/bc3.html

[2] http://www.cs.cmu.edu/~enron/

[3] http://www.csie.ntu.edu.tw/~cjlin/liblinear/

vector machines. The two types of classifier yielded very similar results, with logistic regression classifiers being much faster to train.

3.4 Evaluation Metric

Given the predicted labels on a test set and the existing gold-standard labels of the test set data, in each of our experiments we compute the area under the receiver operator curve as a measure of performance. The area under the ROC (auROC) is a common summary statistic used to measure the quality of binary classification, where a perfect classifier would achieve an auROC of 1.0, and a random classifier, near 0.5.

4 Experiments

4.1 Experimental Design

The available labeled BC3 data totals about 3000 sentences, and the available labeled AMI data totals over 100,000 sentences, so for both efficiency and to not overwhelm the in-domain data, in each of our runs we subsample 10,000 sentences from the AMI data to use for training. After some initial experiments, where increasing the amount of target data beyond this did not improve accuracy, we decided not to incur the runtime cost of training on larger amounts of source data. Similarly, given that we extracted about 200,000 lexical features from our corpora, from our initial experiments trading off auROC and runtime, we decided to select a subset of 10,000 lexical features chosen by having the top mutual information with respect to the summarization labels. We did 5-fold cross-validation to split the target set into training and testing portions, and ran all the domain adaptation methods using the same split. We report the auROC performance of each method averaged over three runs of the 5-fold cross-validation. To test for significant differences between the performances of the various methods, we compute pairwise t-tests between the auROC values obtained on the same run. To account for an increased chance of false positives in reporting results of several pairwise t-tests, we report significance for p-values < 0.005 rather than at the customary 0.05 level.

4.2 Methods Implemented

We compare supervised domain adaptation methods to the baseline INDOMAIN, in which only the training folds of the target data are used for training. In the MERGE method, we simply combine the labeled source and target sets and train on their combination. For ENSEMBLE, we train a classifier on the source training data, a classifier on the target training data, run each of them on the target test data, and for each test instance compute the average of the two probabilities predicted by the classifiers and use it to make a label prediction. We could vary the trade-off between the contribution of the source and target classifier in ENSEMBLE and determine the optimal parameter by cross-validation, though for simplicity we used 0.5 which produced satisfying results. For the PRED approach, we use the source data to train a classifier, use it to make a prediction for the label of each point in the target data, and add the predicted probability as an additional feature to an in-domain trained classifier. The final supervised method FEAT-COPY (Daume, 2007) takes the existing features and extends the feature space by making a general, a source-specific, and a target-specific version of each feature. Hence, a sentence with features (x) gets represented as (x, x, 0) if it comes from the source domain, and as (x, 0, x) if it comes from the target domain.

For semi-supervised domain adaptation methods, our baseline does not exploit any unlabeled target data. We train a classifier on the source data only, and call this TRANSFER. In contrast our two semi-supervised methods try to leverage unlabeled target data to help a classifier trained with labeled source data be more suited to the target domain.

For the SCL approach, we implemented Blitzer's structural correspondence learning (SCL) algorithm. An important part of the algorithm is training a classifier for each of a set of m selected pivot features to determine the correlations of the other features with respect to the pivot. The m models' weights are combined in a matrix, and its SVD with truncation factor of k is then applied to the data to yield k new features for the data, that are added to the existing features. For the larger set of lexical features, we ran SCL with Blitzer's original choice of m=1000 and k=50, but since the computation was extremely time consuming we scale down m to 100. For the tests with conversational features, since the number of features is 24, we picked m=24 and k=24. We also test SCLSMALL, which uses the same algorithm as SCL to find augmented features, except it then uses only these k features to train, not adding them to the original features. This possibility was suggested in (Blitzer 2008).

As a second semi-supervised method, we implemented SELFTRAIN. The standard self-training algorithm we implemented, inspired by

Blum and Mitchell (1998), is to start with a labeled training set T, create a subset of a fixed size of the unlabeled data U, and then iterate training a classifier on T, making a prediction on the data in U, and take the highest-confidence positive *p* predictions and highest-confidence negative *n* predictions from U with their predicted labels to add to T before replenishing U from the rest of the unlabeled data. We picked the size of the subset U as 200, and to select the top *p*=3 and bottom *n*=17 predictions at each step in order to achieve a ratio of summary to total sentences of 15%, which is near to the known ratio of the labels for AMI.

method	indomain	merge	ensem-ble	featcopy	pred	transfer	selftrain	scl	sclsmall
using conversational features									
auROC	0.838	0.747	0.751	0.839	0.838	0.677	0.678	0.663	0.646
time(s)	0.79	2.42	2.64	8.44	5.38	2.08	100.2	52.85	66.74
using lexical features									
auROC	0.623	0.638	0.667	0.615	0.625	0.636	0.636	0.651	0.742
time(s)	4.87	13.64	13.77	78.63	30.99	9.73	448.8	813.7	828.3

Table 1. Performance and time of domain adaptation methods with the two feature sets

5 Results

In our first experiment, we ran all the domain adaptation methods on the data with conversational features; in our second experiment, we did the same on the data with lexical features. We computed the average of the auROCs and running times obtained for each method in each experiment. Table 1 lists the results of the supervised methods MERGE, ENSEMBLE, and FEATCOPY with baseline INDOMAIN, and the semi-supervised methods SELFTRAIN, SCL, and SCLSMALL with baseline TRANSFER.

The best results for supervised methods (and overall) are achieved by FEATCOPY, PRED, and INDOMAIN with the conversational features, with a similar performance that is significantly better than for MERGE and ENSEMBLE. However, for lexical features MERGE and ENSEMBLE beat their performance, with the significant differences from the baseline INDOMAIN being those of ENSEMBLE and FEATCOPY, the latter now being the worst performer.

For the set of lexical features, all semi-supervised methods improve on TRANSFER. In this setting, all of the differences are significant, with SCLSMALL generating a considerable gain of 10%. For the set of conversational features, SELFTRAIN yields an auROC similar to TRANSFER, and the small difference between the two is not significant. Unlike when using lexical features, SCL and SCLSMALL perform significantly worse than TRANSFER, though this is not unexpected. Because it relies on determining correlation between features, we believe that structural correspondence learning is more appropriate in a high rather than low-dimensional feature space.

Figure 1 shows, for each of the methods, a dark grey bar representing the auROC obtained with the set of conversational features next to a lighter grey one for the lexical features. For the supervised methods on the left (INDOMAIN to PRED), the conversational features yield better performance, and this by an absolute ROC difference of more than 5%. However, notice that no method outperform the baseline INDOMAIN. For the semi-supervised methods on the right, the difference in performance between the two feature sets is less marked, although the auROC of SCLSMALL with lexical features is exceptionally larger.

As shown in Table 1, every one of the domain adaptation methods has a higher average time with lexical features than with conversational features. The semi-supervised methods take longer than the fully supervised methods, and this is due to their algorithms involving more steps. Both SCL and SELFTRAIN take minutes instead of seconds to make a prediction, though their running times are more reasonable than with the initial parameter settings we used in preliminary experiments.

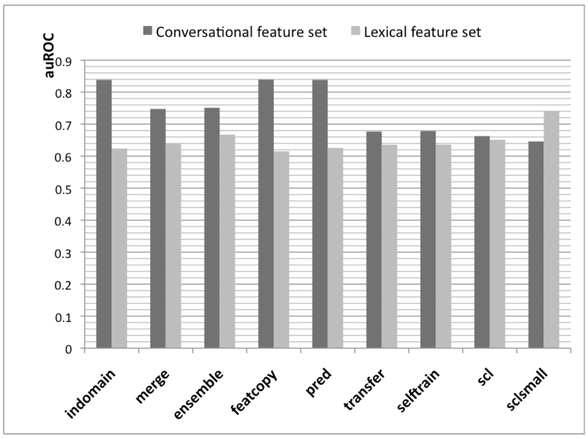

Figure 1. Comparison of auROCs of all domain adaptation methods and baselines

6 Conclusions and Future Work

This paper is a comparative study of the performance of several domain adaptation methods on the task of summarizing conversational data when a large amount of annotated data is available in the domain of meetings and a smaller (or no) amount of annotation exists in the target domain of email threads.

One surprising finding of our experiments is that of the methods we implemented, the best performance is achieved by training on in-domain data using conversational features. Hence, it seems that when sufficient labeled in-domain data is available, supervised domain adaptation is not useful for summarization of emails with the features and amounts of labeled data we used.

However, semi-supervised methods using unlabeled data and labeled out-of-domain data are useful in the absence of these labels, with the SCLSMALL method greatly outperforming the baseline. This is a promising result for using annotated corpora in well-studied domains or conversational modalities to summarize data in new domains.

In our experiments, we have explored the effectiveness of conversational and lexical features separately. The two sets of features differ in their impact on domain adaptation: with conversational features, no method improves significantly over the baseline, whereas with lexical features, the semi-supervised methods given no labeled target data perform better than the supervised baseline of training in-domain. One hypothesis to explain this is that lexical features behave similarly in the two domains, so training on the larger amount of labeled target data is beneficial, while conversational features are more domain specific, likely because emails and meetings are structured differently. As the next step in our work, we intend to combine the two sets of features. In doing this, we will have to ensure that the conversational features are not washed out by a very large number of lexical features.

A scenario of practical interest in domain adaptation for new domains is when the target domain has a considerable amount of unlabeled data and a subset of this data can easily be annotated by hand, for example five threads in the email domain. We are currently exploring injecting a small amount of labeled target data into the semi-supervised methods we have implemented to account for differences that cannot be observed in the unlabeled data. Blitzer (2008) did such an adjustment to SCL using a small amount of labeled target data to correct misaligned features and thus improve accuracy.

Finally, it may be worth investigating how to combine several of the methods, for example by adding the feature of PRED based on training a classifier on the source, alongside augmented features using more unlabeled data through SCL, and adding the highest-confidence labels from SELFTRAIN to the training set.

References

Blitzer, J. (2008). Domain Adaptation of Natural Language Processing Systems. PhD Thesis.

Blitzer, J., Dredze, M., & Pereira, F. (2007). Biographies, Bollywood, Boom-boxes and Blenders: Domain Adaptation for Sentiment Classification. In *Proc. of ACL 2007*.

Blitzer, J., McDonald, R., & Pereira, F. (2006). Domain adaptation with structural correspondence learning. In *Proc. of EMNLP 2006*.

Blum, A., & Mitchell, T. (1998). Combining labeled and unlabeled data with co-training. In *Proc. CLT*.

Carletta, J., Ashby, S., Bourban, S., Flynn, M., Guillemot, M. et al. (2005). The AMI meeting corpus: A pre-announcement. In *Proc. of MLMI 2005*.

Chelba, C., & Acero, A. (2006). Adaptation of maximum entropy capitalizer: Little data can help a lot. *Computer Speech & Language*, *20*(4), 382-399.

Daume III, H. (2007). Frustratingly easy domain adaptation. In *Proc. of ACL 2007*.

Daume III, H., & Marcu, D. (2006). Domain Adaptation for Statistical Classifiers. *Journal of Artificial Intelligence Research*, *26*, 101–126.

Florian, R., Hassan, H., Ittycheriah, A., Jing, H., Kambhatla, N., Luo, X., et al. (2004). A statistical model for multilingual entity detection and tracking. In *Proc. HLT-NAACL 2004*.

Galley, M. (2006). A skip-chain conditional random field for ranking meeting utterances by importance. In *Proc. of EMNLP 2006*.

McClosky, D., Charniak, E., & Johnson, M. (2006). Effective self-training for parsing. In *Proc. of HLT-NAACL 2006* .

Murray, G., & Carenini, G. (2008). Summarizing spoken and written conversations. In *Proc. of EMNLP 2008.*

Murray, G., Carenini, G., & Ng, R. (2010). Interpretation and transformation for abstracting conversations. In *Proc. of HLT-NAACL 2010*.

Murray, G., Renals, S., Moore, J., & Carletta, J. (2006). Incorporating speaker and discourse features into speech summarization. In *Proc. of HLT-NAACL 2006.*

Plank, B. (2009). Structural correspondence learning for parse disambiguation. In *Proc. of EACL 2009: Student Research Workshop.*

Rambow, O., Shrestha, L., & Chen, J. (2004). Summarizing email threads. In *Proc. of HLT-NAACL 2004*.