Discriminative Instance Weighting for Domain Adaptation in Statistical Machine Translation

George Foster and **Cyril Goutte** and **Roland Kuhn**
National Research Council Canada
283 Alexandre-Taché Blvd
Gatineau, QC J8X 3X7
first.last@nrc.gc.ca

Abstract

We describe a new approach to SMT adaptation that weights out-of-domain phrase pairs according to their relevance to the target domain, determined by both how similar to it they appear to be, and whether they belong to general language or not. This extends previous work on discriminative weighting by using a finer granularity, focusing on the properties of instances rather than corpus components, and using a simpler training procedure. We incorporate instance weighting into a mixture-model framework, and find that it yields consistent improvements over a wide range of baselines.

1 Introduction

Domain adaptation is a common concern when optimizing empirical NLP applications. Even when there is training data available in the domain of interest, there is often additional data from other domains that could in principle be used to improve performance. Realizing gains in practice can be challenging, however, particularly when the target domain is distant from the background data. For developers of Statistical Machine Translation (SMT) systems, an additional complication is the heterogeneous nature of SMT components (word-alignment model, language model, translation model, etc.), which precludes a single universal approach to adaptation.

In this paper we study the problem of using a parallel corpus from a background domain (OUT) to improve performance on a target domain (IN) for which a smaller amount of parallel training material—though adequate for reasonable performance—is also available. This is a standard adaptation problem for SMT. It is difficult when IN and OUT are dissimilar, as they are in the cases we study. For simplicity, we assume that OUT is homogeneous. The techniques we develop can be extended in a relatively straightforward manner to the more general case when OUT consists of multiple sub-domains.

There is a fairly large body of work on SMT adaptation. We introduce several new ideas. First, we aim to explicitly characterize examples from OUT as belonging to general language or not. Previous approaches have tried to find examples that are similar to the target domain. This is less effective in our setting, where IN and OUT are disparate. The idea of distinguishing between general and domain-specific examples is due to Daumé and Marcu (2006), who used a maximum-entropy model with latent variables to capture the degree of specificity. Daumé (2007) applies a related idea in a simpler way, by splitting features into general and domain-specific versions. This highly effective approach is not directly applicable to the multinomial models used for core SMT components, which have no natural method for combining split features, so we rely on an instance-weighting approach (Jiang and Zhai, 2007) to downweight domain-specific examples in OUT. Within this framework, we use features intended to capture degree of generality, including the output from an SVM classifier that uses the intersection between IN and OUT as positive examples.

Our second contribution is to apply instance

weighting at the level of phrase pairs. Sentence pairs are the natural instances for SMT, but sentences often contain a mix of domain-specific and general language. For instance, the sentence *Similar improvements in haemoglobin levels were reported in the scientific literature for other epoetins* would likely be considered domain-specific despite the presence of general phrases like *were reported in*. Phrase-level granularity distinguishes our work from previous work by Matsoukas et al (2009), who weight sentences according to sub-corpus and genre membership.

Finally, we make some improvements to baseline approaches. We train linear mixture models for conditional phrase pair probabilities over IN and OUT so as to maximize the likelihood of an empirical joint phrase-pair distribution extracted from a development set. This is a simple and effective alternative to setting weights discriminatively to maximize a metric such as BLEU. A similar maximum-likelihood approach was used by Foster and Kuhn (2007), but for language models only. For comparison to information-retrieval inspired baselines, eg (Lü et al., 2007), we select sentences from OUT using language model perplexities from IN. This is a straightforward technique that is arguably better suited to the adaptation task than the standard method of treating representative IN sentences as queries, then pooling the match results.

The paper is structured as follows. Section 2 describes our baseline techniques for SMT adaptation, and section 3 describes the instance-weighting approach. Experiments are presented in section 4. Section 5 covers relevant previous work on SMT adaptation, and section 6 concludes.

2 Baseline SMT Adaptation Techniques

Standard SMT systems have a hierarchical parameter structure: top-level log-linear weights are used to combine a small set of complex features, interpreted as log probabilities, many of which have their own internal parameters and objectives. The top-level weights are trained to maximize a metric such as BLEU on a small development set of approximately 1000 sentence pairs. Thus, provided at least this amount of IN data is available—as it is in our setting—adapting these weights is straightforward. We focus here instead on adapting the two most important features: the language model (LM), which estimates the probability $p(w|h)$ of a target word w following an ngram h; and the translation models (TM) $p(s|t)$ and $p(t|s)$, which give the probability of source phrase s translating to target phrase t, and vice versa. We do not adapt the alignment procedure for generating the phrase table from which the TM distributions are derived.

2.1 Simple Baselines

The natural baseline approach is to concatenate data from IN and OUT. Its success depends on the two domains being relatively close, and on the OUT corpus not being so large as to overwhelm the contribution of IN.

When OUT is large and distinct, its contribution can be controlled by training separate IN and OUT models, and weighting their combination. An easy way to achieve this is to put the domain-specific LMs and TMs into the top-level log-linear model and learn optimal weights with MERT (Och, 2003). This has the potential drawback of increasing the number of features, which can make MERT less stable (Foster and Kuhn, 2009).

2.2 Linear Combinations

Apart from MERT difficulties, a conceptual problem with log-linear combination is that it multiplies feature probabilities, essentially forcing different features to agree on high-scoring candidates. This is appropriate in cases where it is sanctioned by Bayes' law, such as multiplying LM and TM probabilities, but for adaptation a more suitable framework is often a mixture model in which each event may be generated from some domain. This leads to a linear combination of domain-specific probabilities, with weights in $[0, 1]$, normalized to sum to 1.

Linear weights are difficult to incorporate into the standard MERT procedure because they are "hidden" within a top-level probability that represents the linear combination.[1] Following previous work (Foster and Kuhn, 2007), we circumvent this problem by choosing weights to optimize corpus log-likelihood, which is roughly speaking the training criterion used by the LM and TM themselves.

[1] This precludes the use of exact line-maximization within Powell's algorithm (Och, 2003), for instance.

For the LM, adaptive weights are set as follows:

$$\hat{\alpha} = \operatorname*{argmax}_{\alpha} \sum_{w,h} \tilde{p}(w,h) \log \sum_i \alpha_i p_i(w|h), \quad (1)$$

where α is a weight vector containing an element α_i for each domain (just IN and OUT in our case), p_i are the corresponding domain-specific models, and $\tilde{p}(w,h)$ is an empirical distribution from a target-language training corpus—we used the IN dev set for this.

It is not immediately obvious how to formulate an equivalent to equation (1) for an adapted TM, because there is no well-defined objective for learning TMs from parallel corpora. This has led previous workers to adopt ad hoc linear weighting schemes (Finch and Sumita, 2008; Foster and Kuhn, 2007; Lü et al., 2007). However, we note that the final conditional estimates $p(s|t)$ from a given phrase table maximize the likelihood of joint empirical phrase pair counts over a word-aligned corpus. This suggests a direct parallel to (1):

$$\hat{\alpha} = \operatorname*{argmax}_{\alpha} \sum_{s,t} \tilde{p}(s,t) \log \sum_i \alpha_i p_i(s|t), \quad (2)$$

where $\tilde{p}(s,t)$ is a joint empirical distribution extracted from the IN dev set using the standard procedure.[2]

An alternative form of linear combination is a *maximum a posteriori* (MAP) combination (Bacchiani et al., 2004). For the TM, this is:

$$p(s|t) = \frac{c_I(s,t) + \beta\, p_o(s|t)}{c_I(t) + \beta}, \quad (3)$$

where $c_I(s,t)$ is the count in the IN phrase table of pair (s,t), $p_o(s|t)$ is its probability under the OUT TM, and $c_I(t) = \sum_{s'} c_I(s',t)$. This is motivated by taking $\beta\, p_o(s|t)$ to be the parameters of a Dirichlet prior on phrase probabilities, then maximizing posterior estimates $p(s|t)$ given the IN corpus. Intuitively, it places more weight on OUT when less evidence from IN is available. To set β, we used the same criterion as for α, over a dev corpus:

$$\hat{\beta} = \operatorname*{argmax}_{\beta} \sum_{s,t} \tilde{p}(s,t) \log \frac{c_I(s,t) + \beta\, p_o(s|t)}{c_I(t) + \beta}.$$

[2]Using non-adapted IBM models trained on all available IN and OUT data.

The MAP combination was used for TM probabilities only, in part due to a technical difficulty in formulating coherent counts when using standard LM smoothing techniques (Kneser and Ney, 1995).[3]

2.3 Sentence Selection

Motivated by information retrieval, a number of approaches choose "relevant" sentence pairs from OUT by matching individual source sentences from IN (Hildebrand et al., 2005; Lü et al., 2007), or individual target hypotheses (Zhao et al., 2004). The matching sentence pairs are then added to the IN corpus, and the system is re-trained. Although matching is done at the sentence level, this information is subsequently discarded when all matches are pooled.

To approximate these baselines, we implemented a very simple sentence selection algorithm in which parallel sentence pairs from OUT are ranked by the perplexity of their target half according to the IN language model. The number of top-ranked pairs to retain is chosen to optimize dev-set BLEU score.

3 Instance Weighting

The sentence-selection approach is crude in that it imposes a binary distinction between useful and non-useful parts of OUT. Matsoukas et al (2009) generalize it by learning weights on sentence pairs that are used when estimating relative-frequency phrase-pair probabilities. The weight on each sentence is a value in $[0,1]$ computed by a perceptron with Boolean features that indicate collection and genre membership.

We extend the Matsoukas et al approach in several ways. First, we learn weights on individual phrase pairs rather than sentences. Intuitively, as suggested by the example in the introduction, this is the right granularity to capture domain effects. Second, rather than relying on a division of the corpus into manually-assigned portions, we use features intended to capture the usefulness of each phrase pair. Finally, we incorporate the instance-weighting model into a general linear combination, and learn weights and mixing parameters simultaneously.

[3]Bacchiani et al (2004) solve this problem by reconstituting joint counts from smoothed conditional estimates and unsmoothed marginals, but this seems somewhat unsatisfactory.

3.1 Model

The overall adapted TM is a combination of the form:

$$p(s|t) = \alpha_t \, p_I(s|t) + (1 - \alpha_t) \, p_o(s|t), \quad (4)$$

where $p_I(s|t)$ is derived from the IN corpus using relative-frequency estimates, and $p_o(s|t)$ is an instance-weighted model derived from the OUT corpus. This combination generalizes (2) and (3): we use either $\alpha_t = \alpha$ to obtain a fixed-weight linear combination, or $\alpha_t = c_I(t)/(c_I(t) + \beta)$ to obtain a MAP combination.

We model $p_o(s|t)$ using a MAP criterion over weighted phrase-pair counts:

$$p_o(s|t) = \frac{c_\lambda(s,t) + \gamma u(s|t)}{\sum_{s'} c_\lambda(s',t) + \gamma} \quad (5)$$

where $c_\lambda(s,t)$ is a modified count for pair (s,t) in OUT, $u(s|t)$ is a prior distribution, and γ is a prior weight. The original OUT counts $c_o(s,t)$ are weighted by a logistic function $w_\lambda(s,t)$:

$$\begin{aligned} c_\lambda(s,t) &= c_o(s,t) \; w_\lambda(s,t) \quad (6) \\ &= c_o(s,t) \; [1 + \exp(-\sum_i \lambda_i f_i(s,t))]^{-1}, \end{aligned}$$

where each $f_i(s,t)$ is a feature intended to characterize the usefulness of (s,t), weighted by λ_i.

The mixing parameters and feature weights (collectively ϕ) are optimized simultaneously using dev-set maximum likelihood as before:

$$\hat{\phi} = \operatorname*{argmax}_\phi \sum_{s,t} \tilde{p}(s,t) \log p(s|t;\phi). \quad (7)$$

This is a somewhat less direct objective than used by Matsoukas et al, who make an iterative approximation to expected TER. However, it is robust, efficient, and easy to implement.[4]

To perform the maximization in (7), we used the popular L-BFGS algorithm (Liu and Nocedal, 1989), which requires gradient information. Dropping the conditioning on ϕ for brevity, and letting $\bar{c}_\lambda(s,t) = c_\lambda(s,t) + \gamma u(s|t)$, and $\bar{c}_\lambda(t) = \sum_{s'} \bar{c}_\lambda(s',t)$:

$$\begin{aligned} \frac{\partial \log p(s|t)}{\partial \alpha_t} &= k_t \left[\frac{p_I(s|t)}{p(s|t)} - \frac{p_o(s|t)}{p(s|t)} \right] \\ \frac{\partial \log p(s|t)}{\partial \gamma} &= \frac{1 - \alpha_t}{p(s|t)} \left[\frac{u(s|t)}{\bar{c}_\lambda(t)} - \frac{\bar{c}_\lambda(s,t)}{\bar{c}_\lambda(t)^2} \right] \\ \frac{\partial \log p(s|t)}{\partial \lambda_i} &= \frac{1 - \alpha_t}{p(s|t)} \left[\frac{c_{\lambda'_i}(s,t)}{\bar{c}_\lambda(t)} - \frac{\bar{c}_\lambda(s,t) c_{\lambda'_i}(t)}{\bar{c}_\lambda(t)^2} \right] \end{aligned}$$

where:

$$k_t = \begin{cases} 1 & \text{fixed weight} \\ -c_I(t)/(c_I(t) + \beta)^2 & \text{MAP} \end{cases}$$

$$c_{\lambda'_i}(s,t) = f_i(s,t)(1 - w_\lambda(s,t)) c_\lambda(s,t)$$

and:

$$c_{\lambda'_i}(t) = \sum_{s'} c_{\lambda'_i}(s',t).$$

3.2 Interpretation and Variants

To motivate weighting joint OUT counts as in (6), we begin with the "ideal" objective for setting multinomial phrase probabilities $\theta = \{p(s|t), \forall st\}$, which is the likelihood with respect to the true IN distribution $p_{\hat{I}}(s,t)$. Jiang and Zhai (2007) suggest the following derivation, making use of the true OUT distribution $p_{\hat{o}}(s,t)$:

$$\begin{aligned} \hat{\theta} &= \operatorname*{argmax}_\theta \sum_{s,t} p_{\hat{I}}(s,t) \log p_\theta(s|t) \quad (8) \\ &= \operatorname*{argmax}_\theta \sum_{s,t} \frac{p_{\hat{I}}(s,t)}{p_{\hat{o}}(s,t)} p_{\hat{o}}(s,t) \log p_\theta(s|t) \\ &\approx \operatorname*{argmax}_\theta \sum_{s,t} \frac{p_{\hat{I}}(s,t)}{p_{\hat{o}}(s,t)} c_o(s,t) \log p_\theta(s|t), \end{aligned}$$

where $c_o(s,t)$ are the counts from OUT, as in (6). This has solutions:

$$p_{\hat{\theta}}(s|t) = \frac{p_{\hat{I}}(s,t)}{p_{\hat{o}}(s,t)} c_o(s,t) / \sum_{s'} \frac{p_{\hat{I}}(s',t)}{p_{\hat{o}}(s',t)} c_o(s',t),$$

and from the similarity to (5), assuming $\gamma = 0$, we see that $w_\lambda(s,t)$ can be interpreted as approximating $p_{\hat{I}}(s,t)/p_{\hat{o}}(s,t)$. The logistic function, whose outputs are in $[0,1]$, forces $p_{\hat{I}}(s,t) \leq p_{\hat{o}}(s,t)$. This is not unreasonable given the application to phrase pairs from OUT, but it suggests that an interesting alternative might be to use a plain log-linear weighting

[4] Note that the probabilities in (7) need only be evaluated over the support of $\tilde{p}(s,t)$, which is quite small when this distribution is derived from a dev set. Maximizing (7) is thus much faster than a typical MERT run.

function $\exp(\sum_i \lambda_i f_i(s,t))$, with outputs in $[0,\infty]$. We have not yet tried this.

An alternate approximation to (8) would be to let $w_\lambda(s,t)$ directly approximate $p_{\hat{I}}(s,t)$. With the additional assumption that (s,t) can be restricted to the support of $c_o(s,t)$, this is equivalent to a "flat" alternative to (6) in which each non-zero $c_o(s,t)$ is set to one. This variant is tested in the experiments below.

A final alternate approach would be to combine weighted joint frequencies rather than conditional estimates, ie: $c_I(s,t) + w_\lambda(s,t)c_o(,s,t)$, suitably normalized.[5] Such an approach could be simulated by a MAP-style combination in which separate $\beta(t)$ values were maintained for each t. This would make the model more powerful, but at the cost of having to learn to downweight OUT separately for each t, which we suspect would require more training data for reliable performance. We have not explored this strategy.

3.3 Simple Features

We used 22 features for the logistic weighting model, divided into two groups: one intended to reflect the degree to which a phrase pair belongs to general language, and one intended to capture similarity to the IN domain.

The 14 general-language features embody straightforward cues: frequency, "centrality" as reflected in model scores, and lack of burstiness. They are:

- total number of tokens in the phrase pair (1);
- OUT corpus frequency (1);
- OUT-corpus frequencies of rarest source and target words (2);
- perplexities for OUT IBM1 models, in both directions (2);
- average and minimum source and target word "document frequencies" in the OUT corpus, using successive 100-line pseudo-documents[6] (4); and
- average and minimum source and target word values from the OUT corpus of the following statistic, intended to reflect degree of burstiness (higher values indicate less bursty behaviour): $g/(L - L/(l+1) + \epsilon)$, where g is the sum over all sentences containing the word of the distance (number of sentences) to the nearest sentence that also contains the word, L is the total number of sentences, l is the number of sentences that contain the word, and ϵ is a small constant (4).

The 8 similarity-to-IN features are based on word frequencies and scores from various models trained on the IN corpus:

- 1gram and 2gram source and target perplexities according to the IN LM (4);[7]
- source and target OOV counts with respect to IN (2); and
- perplexities for IN IBM1 models, in both directions (2).

To avoid numerical problems, each feature was normalized by subtracting its mean and dividing by its standard deviation.

3.4 SVM Feature

In addition to using the simple features directly, we also trained an SVM classifier with these features to distinguish between IN and OUT phrase pairs. Phrase tables were extracted from the IN and OUT training corpora (not the dev as was used for instance weighting models), and phrase pairs in the intersection of the IN and OUT phrase tables were used as positive examples, with two alternate definitions of negative examples:

1. Pairs from OUT that are not in IN, but whose source phrase is.
2. Pairs from OUT that are not in IN, but whose source phrase is, and where the intersection of IN and OUT translations for that source phrase is empty.

[5] We are grateful to an anonymous reviewer for pointing this out.

[6] One of our experimental settings lacks document boundaries, and we used this approximation in both settings for consistency.

[7] In the case of the Chinese experiments below, source LMs were trained using text segmented with the LDC segmenter, as were the other Chinese models in our system.

The classifier trained using the 2nd definition had higher accuracy on a development set. We used it to score all phrase pairs in the OUT table, in order to provide a feature for the instance-weighting model.

4 Experiments

4.1 Corpora and System

We carried out translation experiments in two different settings. The first setting uses the European Medicines Agency (EMEA) corpus (Tiedemann, 2009) as IN, and the Europarl (EP) corpus (`www.statmt.org/europarl`) as OUT, for English/French translation in both directions. The dev and test sets were randomly chosen from the EMEA corpus. Figure 1 shows sample sentences from these domains, which are widely divergent.

The second setting uses the news-related subcorpora for the NIST09 MT Chinese to English evaluation[8] as IN, and the remaining NIST parallel Chinese/English corpora (UN, Hong Kong Laws, and Hong Kong Hansard) as OUT. The dev corpus was taken from the NIST05 evaluation set, augmented with some randomly-selected material reserved from the training set. The NIST06 and NIST08 evaluation sets were used for testing. (Thus the domain of the dev and test corpora matches IN.) Compared to the EMEA/EP setting, the two domains in the NIST setting are less homogeneous and more similar to each other; there is also considerably more IN text available.

The corpora for both settings are summarized in table 1.

corpus	sentence pairs
Europarl	1,328,360
EMEA train	11,770
EMEA dev	1,533
EMEA test	1,522
NIST OUT	6,677,729
NIST IN train	2,103,827
NIST IN dev	1,894
NIST06 test	1,664
NIST08 test	1,357

Table 1: Corpora

[8] `www.itl.nist.gov/iad/mig//tests/mt/2009`

The reference medicine for Silapo is EPREX/ERYPO, which contains epoetin alfa.
Le médicament de référence de Silapo est EPREX/ERYPO, qui contient de l'époétine alfa.

—

I would also like to point out to commissioner Liikanen that it is not easy to take a matter to a national court.
Je voudrais préciser, à l'adresse du commissaire Liikanen, qu'il n'est pas aisé de recourir aux tribunaux nationaux.

Figure 1: Sentence pairs from EMEA (top) and Europarl text.

We used a standard one-pass phrase-based system (Koehn et al., 2003), with the following features: relative-frequency TM probabilities in both directions; a 4-gram LM with Kneser-Ney smoothing; word-displacement distortion model; and word count. Feature weights were set using Och's MERT algorithm (Och, 2003). The corpus was word-aligned using both HMM and IBM2 models, and the phrase table was the union of phrases extracted from these separate alignments, with a length limit of 7. It was filtered to retain the top 30 translations for each source phrase using the TM part of the current log-linear model.

4.2 Results

Table 2 shows results for both settings and all methods described in sections 2 and 3. The 1st block contains the simple baselines from section 2.1. The natural baseline (*baseline*) outperforms the pure IN system only for EMEA/EP fren. Log-linear combination (*loglin*) improves on this in all cases, and also beats the pure IN system.

The 2nd block contains the IR system, which was tuned by selecting text in multiples of the size of the EMEA training corpus, according to dev set performance. This significantly underperforms log-linear combination.

The 3rd block contains the mixture baselines. The linear LM (*lin lm*), TM (*lin tm*) and MAP TM (*map tm*) used with non-adapted counterparts perform in all cases slightly worse than the log-linear combination, which adapts both LM and TM components. However, when the linear LM is combined with a

method	EMEA/EP		NIST	
	fren	enfr	nst06	nst08
in	32.77	31.98	27.65	21.65
out	20.42	17.41	19.85	15.71
baseline	33.61	31.15	26.93	21.01
loglin	35.94	32.62	28.09	21.85
ir	33.75	31.91	—	—
lin lm	35.61	31.55	28.02	21.68
lin tm	35.32	32.52	27.16	21.32
map tm	35.15	31.99	27.20	21.17
lm+lin tm	36.42	33.49	27.83	22.03
lm+map tm	36.28	33.31	28.05	22.11
iw all	36.55	33.73	28.74	22.28
iw all map	**37.01**	**33.90**	**30.04**	**23.76**
iw all flat	36.50	33.42	28.31	22.13
iw gen map	36.98	33.75	29.81	23.56
iw sim map	36.82	33.68	29.66	23.53
iw svm map	36.79	33.67	—	—

Table 2: Results, for EMEA/EP translation into English (fren) and French (enfr); and for NIST Chinese to English translation with NIST06 and NIST08 evaluation sets. Numbers are BLEU scores.

linear TM (*lm+lin tm*) or MAP TM (*lm+map TM*), the results are much better than a log-linear combination for the EMEA setting, and on a par for NIST. This is consistent with the nature of these two settings: log-linear combination, which effectively takes the intersection of IN and OUT, does relatively better on NIST, where the domains are broader and closer together. Somewhat surprisingly, there do not appear to be large systematic differences between linear and MAP combinations.

The 4th block contains instance-weighting models trained on all features, used within a MAP TM combination, and with a linear LM mixture. The *iw all map* variant uses a non-0 γ weight on a uniform prior in $p_o(s|t)$, and outperforms a version with $\gamma = 0$ (*iw all*) and the "flattened" variant described in section 3.2. Clearly, retaining the original frequencies is important for good performance, and globally smoothing the final weighted frequencies is crucial. This best instance-weighting model beats the equivalant model without instance weights by between 0.6 BLEU and 1.8 BLEU, and beats the log-linear baseline by a large margin.

The final block in table 2 shows models trained on feature subsets and on the SVM feature described in 3.4. The general-language features have a slight advantage over the similarity features, and both are better than the SVM feature.

5 Related Work

We have already mentioned the closely related work by Matsoukas et al (2009) on discriminative corpus weighting, and Jiang and Zhai (2007) on (non-discriminative) instance weighting. It is difficult to directly compare the Matsoukas et al results with ours, since our out-of-domain corpus is homogeneous; given heterogeneous training data, however, it would be trivial to include Matsoukas-style identity features in our instance-weighting model. Although these authors report better gains than ours, they are with respect to a non-adapted baseline. Finally, we note that Jiang's instance-weighting framework is broader than we have presented above, encompassing among other possibilities the use of unlabelled IN data, which is applicable to SMT settings where source-only IN corpora are available.

It is also worth pointing out a connection with Daumé's (2007) work that splits each feature into domain-specific and general copies. At first glance, this seems only peripherally related to our work, since the specific/general distinction is made for features rather than instances. However, for multinomial models like our LMs and TMs, there is a one to one correspondence between instances and features, eg the correspondence between a phrase pair (s, t) and its conditional multinomial probability $p(s|t)$. As mentioned above, it is not obvious how to apply Daumé's approach to multinomials, which do not have a mechanism for combining split features. Recent work by Finkel and Manning (2009) which re-casts Daumé's approach in a hierarchical MAP framework may be applicable to this problem.

Moving beyond directly related work, major themes in SMT adaptation include the IR (Hildebrand et al., 2005; Lü et al., 2007; Zhao et al., 2004) and mixture (Finch and Sumita, 2008; Foster and Kuhn, 2007; Koehn and Schroeder, 2007; Lü et al., 2007) approaches for LMs and TMs described above, as well as methods for exploiting monolingual in-domain text, typically by translating it automatically and then performing self training (Bertoldi

and Federico, 2009; Ueffing et al., 2007; Schwenk and Senellart, 2009). There has also been some work on adapting the word alignment model prior to phrase extraction (Civera and Juan, 2007; Wu et al., 2005), and on dynamically choosing a dev set (Xu et al., 2007). Other work includes transferring latent topic distributions from source to target language for LM adaptation, (Tam et al., 2007) and adapting features at the sentence level to different categories of sentence (Finch and Sumita, 2008).

6 Conclusion

In this paper we have proposed an approach for instance-weighting phrase pairs in an out-of-domain corpus in order to improve in-domain performance. Each out-of-domain phrase pair is characterized by a set of simple features intended to reflect how useful it will be. The features are weighted within a logistic model to give an overall weight that is applied to the phrase pair's frequency prior to making MAP-smoothed relative-frequency estimates (different weights are learned for each conditioning direction). These estimates are in turn combined linearly with relative-frequency estimates from an in-domain phrase table. Mixing, smoothing, and instance-feature weights are learned at the same time using an efficient maximum-likelihood procedure that relies on only a small in-domain development corpus.

We obtained positive results using a very simple phrase-based system in two different adaptation settings: using English/French Europarl to improve a performance on a small, specialized medical domain; and using non-news portions of the NIST09 training material to improve performance on the news-related corpora. In both cases, the instance-weighting approach improved over a wide range of baselines, giving gains of over 2 BLEU points over the best non-adapted baseline, and gains of between 0.6 and 1.8 over an equivalent mixture model (with an identical training procedure but without instance weighting).

In future work we plan to try this approach with more competitive SMT systems, and to extend instance weighting to other standard SMT components such as the LM, lexical phrase weights, and lexicalized distortion. We will also directly compare with a baseline similar to the Matsoukas et al approach in order to measure the benefit from weighting phrase pairs (or ngrams) rather than full sentences. Finally, we intend to explore more sophisticated instance-weighting features for capturing the degree of generality of phrase pairs.

References

ACL. 2007. *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics (ACL)*, Prague, Czech Republic, June.

Michel Bacchiani, Brian Roark, and Murat Saraclar. 2004. Language model adaptation with MAP estimation and the perceptron algorithm. In NAACL04 (NAA, 2004).

Nicola Bertoldi and Marcello Federico. 2009. Domain adaptation for statistical machine translation with monolingual resources. In WMT09 (WMT, 2009).

Jorge Civera and Alfons Juan. 2007. Domain adaptation in Statistical Machine Translation with mixture modelling. In WMT07 (WMT, 2007).

Hal Daumé III and Daniel Marcu. 2006. Domain Adaptation for Statistical Classifiers. *Journal of Artificial Intelligence Research*, 26:101–126.

Hal Daumé III. 2007. Frustratingly Easy Domain Adaptation. In ACL-07 (ACL, 2007).

Andrew Finch and Eiichiro Sumita. 2008. Dynamic model interpolation for statistical machine translation. In *Proceedings of the ACL Workshop on Statistical Machine Translation*, Columbus, June. WMT.

Jenny Rose Finkel and Christopher D. Manning. 2009. Hierarchical Bayesian domain adaptation. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, Boulder, June. NAACL.

George Foster and Roland Kuhn. 2007. Mixture-model adaptation for SMT. In WMT07 (WMT, 2007).

George Foster and Roland Kuhn. 2009. Stabilizing minimum error rate training. In WMT09 (WMT, 2009).

Almut Silja Hildebrand, Matthias Eck, Stephan Vogel, and Alex Waibel. 2005. Adaptation of the translation model for statistical machine translation based on information retrieval. In *Proceedings of the 10th EAMT Conference*, Budapest, May.

Jing Jiang and ChengXiang Zhai. 2007. Instance Weighting for Domain Adaptation in NLP. In ACL-07 (ACL, 2007).

Reinhard Kneser and Hermann Ney. 1995. Improved backing-off for m-gram language modeling. In *Proceedings of the International Conference on Acoustics,*

Speech, and Signal Processing (ICASSP) 1995, pages 181–184, Detroit, Michigan. IEEE.

Philipp Koehn and Josh Schroeder. 2007. Experiments in domain adaptation for statistical machine translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 224–227, Prague, Czech Republic, June. Association for Computational Linguistics.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 127–133, Edmonton, May. NAACL.

D. C. Liu and J. Nocedal. 1989. On the limited memory method for large scale optimization. *Mathematical Programming B*, 45(3):503–528.

Yajuan Lü, Jin Huang, and Qun Liu. 2007. Improving Statistical Machine Translation Performance by Training Data Selection and Optimization. In *Proceedings of the 2007 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Prague, Czech Republic.

Spyros Matsoukas, Antti-Veikko I. Rosti, and Bing Zhang. 2009. Discriminative corpus weight estimation for machine translation. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Singapore.

NAACL. 2004. *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, Boston, May.

Franz Josef Och. 2003. Minimum error rate training for statistical machine translation. In *Proceedings of the 41th Annual Meeting of the Association for Computational Linguistics (ACL)*, Sapporo, July. ACL.

Holger Schwenk and Jean Senellart. 2009. Translation model adaptation for an arabic/french news translation system by lightly-supervised training. In *Proceedings of MT Summit XII*, Ottawa, Canada, September. International Association for Machine Translation.

Yik-Cheung Tam, Ian Lane, and Tanja Schultz. 2007. Bilingual-LSA Based LM Adaptation for Spoken Language Translation. In ACL-07 (ACL, 2007).

Jorg Tiedemann. 2009. News from opus - a collection of multilingual parallel corpora with tools and interfaces. In N. Nicolov, K. Bontcheva, G. Angelova, and R. Mitkov, editors, *Recent Advances in Natural Language Processing*, volume V, pages 237–248. John Benjamins, Amsterdam/Philadelphia.

Nicola Ueffing, Gholamreza Haffari, and Anoop Sarkar. 2007. Transductive learning for statistical machine translation. In ACL-07 (ACL, 2007).

WMT. 2007. *Proceedings of the ACL Workshop on Statistical Machine Translation*, Prague, June.

WMT. 2009. *Proceedings of the 4th Workshop on Statistical Machine Translation*, Athens, March.

Hua Wu, Haifeng Wang, and Zhanyi Liu. 2005. Alignment model adaptation for domain-specific word alignment. In *Proceedings of the 43th Annual Meeting of the Association for Computational Linguistics (ACL)*, Ann Arbor, Michigan, July. ACL.

Jia Xu, Yonggang Deng, Yuqing Gao, and Hermann Ney. 2007. Domain dependent statistical machine translation. In *MT Summit XI*, Copenhagen, September.

Bing Zhao, Matthias Eck, and Stephan Vogel. 2004. Language model adaptation for statistical machine translation with structured query models. In *Proceedings of the International Conference on Computational Linguistics (COLING) 2004*, Geneva, August.