Adapting to All Domains at Once: Rewarding Domain Invariance in SMT

Hoang Cuong and **Khalil Sima'an** and **Ivan Titov**
Institute for Logic, Language and Computation
University of Amsterdam
Science Park 107, 1098 XG Amsterdam, The Netherlands
{c.hoang,k.simaan,titov}@uva.nl

Abstract

Existing work on domain adaptation for statistical machine translation has consistently assumed access to a small sample from the test distribution (target domain) at training time. In practice, however, the target domain may not be known at training time or it may change to match user needs. In such situations, it is natural to push the system to make safer choices, giving higher preference to domain-invariant translations, which work well across domains, over risky domain-specific alternatives. We encode this intuition by (1) inducing latent subdomains from the training data only; (2) introducing features which measure how specialized phrases are to individual induced sub-domains; (3) estimating feature weights on out-of-domain data (rather than on the target domain). We conduct experiments on three language pairs and a number of different domains. We observe consistent improvements over a baseline which does not explicitly reward domain invariance.

1 Introduction

Mismatch in phrase translation distributions between test data (*target domain*) and train data is known to harm performance of statistical translation systems (Irvine et al., 2013; Carpuat et al., 2014). Domain-adaptation methods (Foster et al., 2010; Bisazza et al., 2011; Sennrich, 2012b; Razmara et al., 2012; Sennrich et al., 2013; Haddow, 2013; Joty et al., 2015) aim to specialize a system estimated on out-of-domain training data to a target domain represented by a small data sample. In practice, however, the target domain may not be known at training time or it may change over time depending on user needs. In this work we address exactly the setting where we have a domain-agnostic system but we have no access to any samples from the target domain at training time. This is an important and challenging setting which, as far as we are aware, has not yet received attention in the literature.

When the target domain is unknown at training time, the system could be trained to make safer choices, preferring translations which are likely to work across different domains. For example, when translating from English to Russian, the most natural translation for the word 'code' would be highly dependent on the domain (and the corresponding word sense). Russian words 'шифр', 'закон' or 'программа' would perhaps be optimal choices if we consider cryptography, legal and software development domains, respectively. However, the translation 'код' is also acceptable across all these domains and, as such, would be a safer choice when the target domain is unknown. Note that such a translation may not be the most frequent overall and, consequently, might not be proposed by a standard (i.e. domain-agnostic) phrase-based translation system.

In order to encode preference for domain-invariant translations, we introduce a measure which quantifies how likely a phrase (or a phrase-pair) is to be "domain-invariant". We recall that most large parallel corpora are heterogeneous, consisting of diverse language use originating from a variety of unspecified subdomains. For example, news articles may cover sports, finance, politics, technology and a variety of other news topics. None of the subdomains may match the target domain particularly

well, but they can still reveal how domain-specific a given phrase is. For example, if we would observe that the word 'code' can be translated as 'код' across cryptography and legal subdomains observed in training data, we can hypothesize that it may work better on a new unknown domain than 'закон' which was specific only to a single subdomain (legal). This would be a suitable decision if the test domain happens to be software development, even though no texts pertaining to this domain were included in the heterogeneous training data.

Importantly, the subdomains are usually not specified in the heterogeneous training data. Therefore, we treat the subdomains as latent, so we can induce them automatically. Once induced, we define measures of domain specificity, particularly expressing two generic properties:

Phrase domain specificity How specific is a target or a source phrase to some of the induced subdomains?

Phrase pair domain coherence How coherent is a source phrase and a target language translation across the induced subdomains?

These features capture two orthogonal aspects of phrase behaviour in heterogeneous corpora, with the rationale that phrase pairs can be weighted along these two dimensions. *Domain-specificity* captures the intuition that the more specific a phrase is to certain subdomains, the less applicable it is in general. Note that specificity is applied not only to target phrases (as 'код' and 'закон' in the above example) but also to source phrases. When applied to a source phrase, it may give a preference towards using shorter phrases as they are inherently less domain specific. In contrast to phrase domain specificity, *phrase pair coherence* reflects whether candidate target and source phrases are typically used in the same set of domains. The intuition here is that the more divergent the distributional behaviour of source and target phrases across subdomains, the less certain we are whether this phrase pair is valid for the unknown target domain. In other words, a translation rule with source and target phrases having two similar distributions over the latent subdomains is likely safer to use.

Weights for these features, alongside all other standard features, are tuned on a development set. Importantly, we show that there is no noteworthy benefit from tuning the weights on a sample from the target domain. It is enough to tune them on a mixed-domain dataset sufficiently different from the training data. We attribute this attractive property to the fact that our features, unlike the ones typically considered in standard domain-adaptation work, are generic and only affect the amount of risk our system takes. In contrast, for example, in (Eidelman et al., 2012; Chiang et al., 2011; Hu et al., 2014; Hasler et al., 2014; Su et al., 2015; Sennrich, 2012b; Chen et al., 2013b; Carpuat et al., 2014), features capture similarities between a target domain and each of the training subdomains. Clearly, domain adaptation with such rich features, though potentially more powerful, would not be possible without a development set closely matching the target domain.

We conduct our experiments on three language pairs and explore adaptation to 9 domain adaptation tasks in total. We observe significant and consistent performance improvements over the baseline domain-agnostic systems. This result confirms that our two features, and the latent subdomains they are computed from, are useful also for the very challenging domain adaptation setting considered in this work.

2 Domain-Invariance for Phrases

At the core of a standard state-of-the-art phrase-based system (Koehn et al., 2003; Och and Ney, 2004) lies a phrase table $\{\langle \tilde{e}, \tilde{f} \rangle\}$ extracted from a word-aligned training corpus together with estimates for phrase translation probabilities $P_{count}(\tilde{e}|\,\tilde{f})$ and $P_{count}(\tilde{f}|\,\tilde{e})$. Typically the phrases and their probabilities are obtained from large parallel corpora, which are usually broad enough to cover a mixture of several subdomains. In such mixtures, phrase distributions may be different across different subdomains. Some phrases (whether source or target) are more specific for certain subdomains than others, while some phrases are useful across many subdomains. Moreover, for a phrase pair, the distribution over the subdomains for its source side may be similar or not to the distribution for its target side. Coherent pairs seem safer to employ than pairs

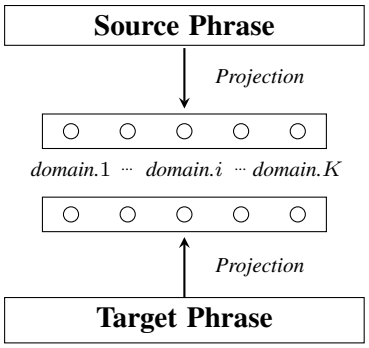

Figure 1: The projection framework of phrases into K-dimensional vector space of probabilistic latent subdomains.

which exhibit different distributions over the subdomains. These two factors, domain specificity and domain coherence, can be estimated from the training corpus if we have access to subdomain statistics for the phrases. In the setting addressed here, the subdomains are not known in advance and we have to consider them latent in the training data.

Therefore, we introduce a random variable $z \in \{1, \ldots, K\}$ encoding (arbitrary) K latent subdomains that generate each source and target phrase $\tilde{e}$ and $\tilde{f}$ of every phrase pair $\langle \tilde{e}, \tilde{f} \rangle$. In the next Section, we aim to estimate distributions $P(z|\ \tilde{e})$ and $P(z|\ \tilde{f})$ for subdomain z over the source and target phrases respectively. In other words, we aim at *projecting* phrases onto a compact $(K-1)$ dimensional simplex of subdomains with vectors:

$$\vec{\tilde{e}} = \Big[P(z=1|\ \tilde{e}),\ \ldots,\ P(z=K|\ \tilde{e})\Big], \quad (1)$$

$$\vec{\tilde{f}} = \Big[P(z=1|\ \tilde{f}),\ \ldots,\ P(z=K|\ \tilde{f})\Big]. \quad (2)$$

Each of the K elements encodes how well each source and target phrase expresses a specific latent subdomain in the training data. See Fig. 1 for an illustration of the projection framework. Once the projection is performed, the hidden cross-domain translation behaviour of phrases and phrase pairs can be modeled as follows:

- *Domain-specificity of phrases*: A rule with source and target phrases having a *peaked* distribution over latent subdomains is likely domain-specific. Technically speaking, entropy comes as a natural choice for quantifying domain specificity. Here, we opt for the Renyi entropy and define the domain specificity as follows:

$$D_\alpha(\vec{\tilde{e}}) = \frac{1}{1-\alpha} \log \Big(\sum_{i=1}^{K} P(z=i|\ \tilde{e})^\alpha \Big)$$

$$D_\alpha(\vec{\tilde{f}}) = \frac{1}{1-\alpha} \log \Big(\sum_{i=1}^{K} P(z=i|\ \tilde{f})^\alpha \Big)$$

  For convenience, we refer to $D_\alpha(\cdot)$ as the domain specificity of a phrase. In this study, we choose the value of α as 2 which is the default choice (also known as the Collision entropy).

- *Source-target coherence across subdomains*: A translation rule with source and target phrases having two similar distributions over the latent subdomains is likely safer to use. We use the Chebyshev distance for measuring the similarity between two distributions. The divergence of two vectors $\vec{\tilde{e}}$ and $\vec{\tilde{f}}$ is defined as follows

$$D(\vec{\tilde{e}},\ \vec{\tilde{f}}) = \max_{i=\{1,\ldots,K\}} \Big| P(z=i|\tilde{e}) - P(z=i|\tilde{f}) \Big|$$

  We refer to $D(\vec{\tilde{e}}, \vec{\tilde{f}})$ as the phrase pair coherence across latent subdomains.

We investigated some other similarities for phrase pair coherence (the Kullback-Leibler divergence and the Hellinger distance) but have not observed any noticeable improvements in the performance. We will discuss these experiments in the empirical section.

Once computed for every phrase pair, the two measures $D_\alpha(\vec{\tilde{e}})$, $D_\alpha(\vec{\tilde{f}})$ $D(\vec{\tilde{e}},\ \vec{\tilde{f}})$, will be integrated into a phrase-based SMT system as feature functions.

3 Latent Subdomain Induction

We now present our approach for inducing *latent subdomain distributions* $P(z|\ \tilde{e})$ and $P(z|\ \tilde{f})$ for every source and target phrases $\tilde{e}$ and $\tilde{f}$. In our experiments, we compare using our subdomain induction framework with relying on topic distributions provided by a standard topic model, Latent Dirichlet Allocation (Blei et al., 2003). Note that unlike LDA we rely on parallel data and word alignments when inducing domains. Our intuition is that latent variables capturing regularities in bilingual data may be more appropriate for the translation task.

Inducing these probabilities directly is rather difficult as the task of designing a fully generative phrase-based model is known to be challenging.[1] In order to avoid this, we follow Matsoukas et al. (2009) and Cuong and Sima'an (2014a) who "embed" such a phrase-level model into a latent subdomain model that works at the *sentence level*. In other words, we associate latent domains with sentence pairs rather than with phrases, and use the posterior probabilities computed for the sentences with all the phrases appearing in the corresponding sentences. Given $P(z|\ \mathbf{e},\ \mathbf{f})$ - a latent subdomain model given sentence pairs $\langle \mathbf{e}, \mathbf{f} \rangle$ - the estimation of $P(z|\ \tilde{e})$ and $P(z|\ \tilde{f})$, for phrases $\tilde{e}$ and $\tilde{f}$, can be simplified by computing expectations z for all $z \in \{1,\ \ldots,\ K\}$:

$$P(z = i|\tilde{e}) = \frac{\sum_{\mathbf{e},\mathbf{f}} P(z = i|\mathbf{e},\mathbf{f}) c(\tilde{e};\mathbf{e})}{\sum_{i'=1}^{K} \sum_{\mathbf{e},\mathbf{f}} P(z = i'|\mathbf{e},\mathbf{f}) c(\tilde{e};\mathbf{e})},$$

$$P(z = i|\tilde{f}) = \frac{\sum_{\mathbf{e},\mathbf{f}} P(z = i|\mathbf{e},\mathbf{f}) c(\tilde{f};\mathbf{f})}{\sum_{i'=1}^{K} \sum_{\mathbf{e},\mathbf{f}} P(z = i'|\mathbf{e},\mathbf{f}) c(\tilde{f};\mathbf{f})}.$$

Here, $c(\tilde{e}, \mathbf{e})$ is the count of a phrase $\tilde{e}$ in a sentence $\mathbf{e}$ in the training corpus.

Latent subdomains for sentences. We now turn to describing our latent subdomain model for sentences. We assume the following generative story for sentence pairs:

1. generate the domain z from the prior $P(z)$;
2. choose the generation direction: $\mathbf{f}$-to-$\mathbf{e}$ or $\mathbf{e}$-to-$\mathbf{f}$, with equal probability;
3. if the $\mathbf{e}$-to-$\mathbf{f}$ direction is chosen then generate the pair relying on $P(\mathbf{e}|\ z)P(\mathbf{f}|\ \mathbf{e},\ z)$;
4. otherwise, use $P(\mathbf{f}|\ z)P(\mathbf{e}|\ \mathbf{f},\ z)$.

Formally, it is a uniform mixture of the generative processes for the two potential translation directions.[2] This generative story implies having two translation models (TMs) and two language models (LMs), each augmented with latent subdomains. Now, the posterior $P(z|\ \mathbf{e},\ \mathbf{f})$ can be computed as

$$P(z|\ \mathbf{e},\ \mathbf{f}) \propto P(z)\Big(\frac{1}{2}P(\mathbf{e}|\ z)P(\mathbf{f}|\ \mathbf{e},\ z) + \frac{1}{2}P(\mathbf{f}|\ z)P(\mathbf{e}|\ \mathbf{f},\ z)\Big). \quad (3)$$

As we aim for a simple approach, our TMs are computed through the introduction of hidden alignments $\mathbf{a}$ and $\mathbf{a}'$ in $\mathbf{f}$-to-$\mathbf{e}$ and $\mathbf{e}$-to-$\mathbf{f}$ directions respectively, in which $P(\mathbf{f}|\ \mathbf{e},\ z) = \sum_{\mathbf{a}} P(\mathbf{f},\ \mathbf{a}|\ \mathbf{e},\ z)$ and $P(\mathbf{e}|\ \mathbf{f},\ z) = \sum_{\mathbf{a}'} P(\mathbf{e},\ \mathbf{a}'|\ \mathbf{f},\ z)$. To make the marginalization of alignments tractable, we restrict $P(\mathbf{f},\ \mathbf{a}|\ \mathbf{e},\ z)$ and $P(\mathbf{e},\ \mathbf{a}'|\ \mathbf{f},\ z)$ to the same assumptions as IBM Model 1 (Brown et al., 1993) (i.e. a multiplication of translation of lexical probabilities with respect to latent subdomains).

We use standard n^{th}-order Markov model for $P(\mathbf{e}|\ z)$ and $P(\mathbf{f}|\ z)$, in which $P(\mathbf{e}|\ z) = \prod_i P(e_i|\ e_{i-n}^{i-1},\ z)$ and $P(\mathbf{f}|\ z) = \prod_j P(f_j|\ f_{j-n}^{j-1},\ z)$. Here, the notation e_{i-n}^{i-1} and f_{j-n}^{j-1} is used to denote the history of length n for the source and target words e_i and f_j, respectively.

Training. For training, we maximize the log-likelihood $\mathcal{L}$ of the data

$$\mathcal{L} = \sum_{\mathbf{e},\mathbf{f}} \log\Big(\sum_{z} P(z)\Big(\frac{1}{2}P(\mathbf{e}|z)\sum_{\mathbf{a}} P(\mathbf{f},\mathbf{a}|\mathbf{e},z) + \frac{1}{2}P(\mathbf{f}|\ z)\sum_{\mathbf{a}'} P(\mathbf{e},\mathbf{a}'|\ \mathbf{f},\ z)\Big)\Big). \quad (4)$$

As there is no closed-form solution, we use the expectation-maximization (EM) algorithm (Dempster et al., 1977).

In the **E**-step, we compute the posterior distributions $P(\mathbf{a},\ z|\ \mathbf{e},\ \mathbf{f})$ and $P(\mathbf{a}',\ z|\ \mathbf{e},\ \mathbf{f})$ as follows

$$P(\mathbf{a},\ z|\ \mathbf{e},\mathbf{f}) \propto P(z)\Big(P(\mathbf{e}|\ z)P(\mathbf{f},\ \mathbf{a}|\ \mathbf{e},\ z) + P(\mathbf{f}|\ z)\sum_{\mathbf{a}'} P(\mathbf{e},\ \mathbf{a}'|\ \mathbf{f},\ z)\Big), \quad (5)$$

$$P(\mathbf{a}',\ z|\ \mathbf{e},\mathbf{f}) \propto P(z)\Big(P(\mathbf{e}|\ z)\sum_{\mathbf{a}} P(\mathbf{f},\ \mathbf{a}|\ \mathbf{e},\ z) + P(\mathbf{f}|\ z)P(\mathbf{e},\ \mathbf{a}'|\ \mathbf{f},\ z)\Big). \quad (6)$$

[1] Doing that requires incorporating into the model additional hidden variables encoding phrase segmentation (DeNero et al., 2006). This would significantly complicate inference (Mylonakis and Sima'an, 2008; Neubig et al., 2011; Cohn and Haffari, 2013).

[2] Note that we effectively average between them which is reasonable, as there is no reason to give preference to any of them.

In the **M**-step, we use the posteriors $P(\mathbf{a},\ z|\ \mathbf{e},\ \mathbf{f})$ and $P(\mathbf{a}',\ z|\ \mathbf{e},\ \mathbf{f})$ to re-estimate parameters of both alignment models. This is done in a very similar way to estimation of the standard IBM Model 1.

We use the posteriors to re-estimate LM parameters as follows

$$P(e_i|e_1^{i-1}, z) \propto \sum_{\mathbf{e},\mathbf{f}} P(z|\mathbf{e},\mathbf{f})c(e_1^i;\ \mathbf{e}), \quad (7)$$

$$P(f_i|f_1^{i-1}, z) \propto \sum_{\mathbf{e},\mathbf{f}} P(z|\mathbf{e},\mathbf{f})c(f_1^i;\ \mathbf{f}). \quad (8)$$

To obtain better parameter estimates for word predictions and avoid overfitting, we use smoothing in the M-step. In this work, we chose to apply *expected Kneser-Ney smoothing* technique (Zhang and Chiang, 2014) as it is simple and achieves state-of-the-art performance on the language modeling problem.

Finally, $P(z)$ can be simply estimated as follows

$$P(z) \propto \sum\nolimits_{\mathbf{e},\ \mathbf{f}} P(z|\ \mathbf{e},\ \mathbf{f}) \quad (9)$$

Hierarchical Training. In practice, we found that training the full joint model leads to brittle performance, as EM is very likely to get stuck in bad local maxima. To address this difficulty, in our implementation, we start out by first jointly training $P(z)$, $P(\mathbf{e}|\ z)$ and $P(\mathbf{f}|\ z)$. In this way in the E-step, we fix our model parameters and compute $P(z|\ \mathbf{e},\ \mathbf{f})$ for every sentence pair: $P(z|\ \mathbf{e},\ \mathbf{f}) \propto P(\mathbf{e}|\ z)P(\mathbf{f}|\ z)P(z)$. In the M-step, we use the posteriors to re-estimate the model parameters, as in Equations (7), (8) and (9). Once the model is trained, we fix the language modeling parameters and finally train the full model.

This parallel latent subdomain language model is less expressive and, consequently, is less likely to get stuck in a local maximum. The LMs estimated in this way will then drive the full alignment model towards better configurations in the parameter space.[3] In practice, this training scheme is particularly useful in case of learning a more fine-grained latent subdomain model with larger K.

[3]This procedure can be regarded as a form of hierarchical estimation: we start with a simpler model and then use it to drive a more expressive model. Note that we also use $P(z)$ estimated within the parallel latent subdomain LMs to initialize $P(z)$ for the latent subdomain alignment model.

4 Experiments

Training Data		**English**	**French**
	Sents	**5.01M**	
	Words	103.39M	125.81M
		English	**Spanish**
	Sents	**4.00M**	
	Words	81.48M	89.08M
		English	**German**
	Sents	**4.07M**	
	Words	93.19M	88.48M

Table 1: Data Preparation.

4.1 Data

We conduct experiments with large-scale SMT systems across a number of domains for three language pairs (*English-Spanish*, *English-German* and *English-French*). The datasets are summarized in Table 1. For *English-Spanish*, we run experiments with training data consisting of $4M$ sentence pairs collected from multiple resources within the WMT 2013 MT Shared Task. These include EuroParl (Koehn, 2005), Common Crawl Corpus, UN Corpus, and News Commentary. For *English-German*, our training data consists of $4.1M$ sentence pairs collected from the WMT 2015 MT Shared Task, including EuroParl, Common Crawl Corpus and News Commentary. Finally, for *English-French*, we train SMT systems on a corpus of $5M$ sentence pairs collected from the WMT 2015 MT Shared Task, including the 109 French-English corpus.

We conducted experiments on 9 different domains (tasks) where the data was manually collected by a TAUS.[4] Table 2 presents the translation tasks: each of the tasks deals with a specific domain, each of this task has presumably a very different relevance level to the training data. In this way, we test the stability of our results across a wide range of target domains.

4.2 Systems

We use a standard state-of-the-art phrase-based system. The Baseline system includes MOSES (Koehn et al., 2007) baseline feature functions, plus eight hierarchical lexicalized reordering model feature functions (Galley and Manning, 2008). The training data is first word-aligned using GIZA++ (Och and

[4]`https://www.taus.net/`.

			English	French
Professional &Business Services	Dev	Sents	$2K$	
		Words	74.16K	83.85K
	Test	Sents	$5K$	
		Words	92.84K	105.05K
Leisure, Tourism and Arts	Dev	Sents	$2K$	
		Words	107.45K	117.16K
	Test	Sents	$5K$	
		Words	101.82K	114.76K
			English	**Spanish**
Professional &Business Services	Dev	Sents	2K	
		Words	31.70K	34.62K
	Test	Sents	5K	
		Words	84.1K	93.4K
Legal	Dev	Sents	2K	
		Words	35.06K	38.78K
	Test	Sents	5K	
		Words	88.63K	102.71K
Financials	Dev	Sents	2K	
		Words	37.23K	42.89K
	Test	Sents	5K	
		Words	99.05K	109.81K
			English	**German**
Professional &Business Services	Dev	Sents	$2K$	
		Words	80.49K	85.08K
	Test	Sents	$5K$	
		Words	79.75K	85.28K
Legal	Dev	Sents	$2K$	
		Words	50.54K	45.99K
	Test	Sents	$5K$	
		Words	124.93K	111.70K
Computer Software	Dev	Sents	$2K$	
		Words	40.24K	38.31K
	Test	Sents	$5K$	
		Words	102.71K	101.12K
Computer Hardware	Dev	Sents	$2K$	
		Words	37.40K	36.98K
	Test	Sents	$5K$	
		Words	103.29K	98.04K

Table 2: Data and adaptation tasks.

Ney, 2003) and then symmetrized with *grow(-diag)-final-and* (Koehn et al., 2003). We limit the phrase length to the maximum of seven words. The language models are interpolated 5-grams with Kneser-Ney smoothing, estimated by KenLM (Heafield et al., 2013) from a large monolingual corpus of nearly $2.1B$ English words collected within the WMT 2015 MT Shared Task. Finally, we use MOSES as a decoder (Koehn et al., 2007).

Our system is exactly the same as the baseline, plus three additional feature functions induced for the translation rules: two features for *domain-specificity of phrases* (both for the source side ($D_\alpha(\vec{\tilde{f}})$) and the target side ($D_\alpha(\vec{\tilde{e}})$), and one feature for *source-target coherence across subdomains* ($D(\vec{\tilde{e}}, \vec{\tilde{f}})$). For the projection, we use K=12. We also explored different values for K, but have not observed significant difference in the scores. In our experiments we do one iteration of EM with parallel LMs (as described in Section 3), before continuing with the full model for three more iterations. We did not observe a significant improvement from running EM any longer. Finally, we use *hard EM*, as it has been found to yield better models than the standard soft EM on a number of different task (e.g., (Johnson, 2007)). In other words, instead of standard 'soft' EM updates with phrase counts weighted according to the posterior $P(z = i|\ \mathbf{e},\ \mathbf{f})$, we use the 'winner-takes-all' approach:

$$P(z\ =\ i|\ \tilde{e}) \propto \sum_{\langle \mathbf{e},\mathbf{f}\rangle} c(i;\ \hat{z}_{\langle \mathbf{e},\mathbf{f}\rangle})\delta(\tilde{e};\ \mathbf{e}),$$

$$P(z\ =\ i|\ \tilde{f}) \propto \sum_{\langle \mathbf{e},\mathbf{f}\rangle} c(i;\ \hat{z}_{\langle \mathbf{e},\mathbf{f}\rangle})\delta(\tilde{f};\ \mathbf{f}).$$

Here, $\hat{z}_{\langle \mathbf{e},\ \mathbf{f}\rangle}$ is the "winning" latent subdomain for sentence pair $\langle \mathbf{e},\ \mathbf{f}\rangle$:

$$\hat{z}_{\langle \mathbf{e},\ \mathbf{f}\rangle} = \underset{i \in \{1,\ ...,\ K\}}{\operatorname{argmax}}\ P(z = i|\ \mathbf{e},\ \mathbf{f})$$

In practice, we found that using this hard version leads to better performance.[5]

4.3 Alternative tuning scenarios

In order to tune all systems, we use the k-best batch MIRA (Cherry and Foster, 2012). We report the translation accuracy with three metrics - BLEU (Papineni et al., 2002), METEOR (Denkowski and Lavie, 2011) and TER (Snover et al., 2006). We mark an improvement as significant when we obtain the p-level of 5 % under paired bootstrap resampling (Koehn, 2004). Note that better results correspond to larger BLEU and METEOR but to smaller TER. For every system reported, we run the optimizer at least three times, before running MultEval (Clark et al., 2011) for resampling and significance testing. Note that the scores for the systems are averages over multiple runs.

[5] A more principled alternative would be to use posterior regularization (Ganchev et al., 2009).

Task	System	BLEU↑ /Δ	METEOR↑ /Δ	TER↓ /Δ
English-French				
Professional & Business Services	Baseline	21.4	28.8	60.0
	Our System	21.5/+0.1	28.9/**+0.1**	59.7/**-0.3**
Leisure, Tourism and Arts	Baseline	39.9	36.7	48.1
	Our System	40.8/**+0.9**	37.1/**+0.4**	47.1/**-1.0**
English-Spanish				
Financials	Baseline	32.5	37.1	45.6
	Our System	32.8/**+0.3**	37.2/**+0.1**	45.4/**-0.2**
Professional & Business Services	Baseline	24.4	31.7	54.9
	Our System	24.8/**+0.4**	31.9/**+0.2**	54.8/-0.1
Legal Services	Baseline	33.3	36.3	49.5
	Our System	33.8/**+0.5**	36.5/**+0.2**	49.1/**-0.4**
English-German				
Computer Software	Baseline	22.8	27.7	64.3
	Our System	23.1/**+0.3**	27.8/**+0.1**	64.0/**-0.3**
Computer Hardware	Baseline	20.5	27.7	61.2
	Our System	20.9/**+0.4**	27.9/**+0.2**	61.1/**-0.1**
Professional & Business Services	Baseline	15.3	25.4	69.2
	Our System	15.7/**+0.4**	25.6/**+0.2**	68.6/**-0.6**
Legal Services	Baseline	29.6	32.9	55.6
	Our System	30.2/**+0.6**	33.3/**+0.4**	55.1/**-0.5**

Table 3: Adaptation results when tuning on the **in-domain** development set. The bold face indicates that the improvement over the baseline is significant.

Task	System	BLEU↑ /Δ	METEOR↑ /Δ	TER↓ /Δ
English-French				
Professional & Business Services	Baseline	20.7	28.3	59.5
	Our System	20.7/+0.0	28.4/**+0.1**	59.4/-0.1
Leisure, Tourism and Arts	Baseline	39.7	37.0	48.6
	Our System	40.6/**+0.9**	37.4/**+0.4**	47.4/**-1.2**
English-Spanish				
Financials	Baseline	33.6	37.5	45.4
	Our System	34.0/**+0.4**	37.7/**+0.2**	45.0/**-0.4**
Professional & Business Services	Baseline	24.4	31.9	55.3
	Our System	24.9/**+0.5**	32.0/**+0.1**	54.9/**-0.4**
Legal Services	Baseline	32.4	35.8	49.0
	Our System	32.9/**+0.5**	36.0/**+0.2**	48.8/**-0.2**
English-German				
Computer Software	Baseline	23.2	27.6	63.4
	Our System	23.5/**+0.3**	27.8/**+0.2**	63.0/**-0.4**
Computer Hardware	Baseline	20.8	27.8	61.5
	Our System	21.0/**+0.2**	28.0/**+0.2**	61.2/**-0.3**
Professional & Business Services	Baseline	13.8	25.2	72.2
	Our System	13.9/**+0.1**	25.3/**+0.1**	72.1/**-0.1**
Legal Services	Baseline	29.3	32.7	55.2
	Our System	29.9/**+0.6**	33.1/**+0.4**	54.6/**-0.6**

Table 4: Adaptation results when tuning on the **mixed-domain** development set. The bold face indicates that the improvement over the baseline is significant.

For tuning the systems we explore two kinds of development sets: (1) An **in-domain** development set of in-domain data that directly exemplifies the translation task (i.e. a sample of target-domain data), and (2) a **mixed-domain** development set which is a *full* concatenation of development sets from all the available domains for a language pair; this scenario is a more realistic one when no in-domain data is

available. In the analysis section we also test these two scenarios against the scenario **mixed-domain minus in-domain**, which excludes the in-domain development set part from the mixed-domain development set. By exploring the three different development sets we hope to shed light on the importance of having samples from the target domain when using our features. If our features can indeed capture domain invariance of phrases then they should improve the performance in all three settings, including the most difficult setting where the in-domain data has been explicitly excluded from the tuning phase.

4.4 Main results

In-domain tuning scenario. Table 3 presents the results for the in-domain development set scenario. The integration of the domain-invariant feature functions into the baseline results in a significant improvement across all domains: average $+0.50$ BLEU on two adaptation tasks for English-French, $+0.40$ BLEU on three adaptation tasks for English-Spanish and $+0.43$ BLEU on four adaptation tasks for English-German.

Mixed-domain tuning scenario. While the improvements are robust and consistent for the *in-domain* development set scenario, we are especially delighted to see a similar improvement for the mixed-domain tuning scenario (Table 4). In detail, we observe an average $+0.45$ BLEU on two adaptation tasks for English-French, $+0.47$ BLEU on three adaptation tasks for English-Spanish and $+0.30$ BLEU on four adaptation tasks for English-German. We would like to emphasize that this performance improvement is obtained without tuning specifically for the target domain or using other domain-related meta-information in the training corpus.

Additional analysis

We investigate the individual contribution of each domain-invariance feature. We conduct experiments using a basic large-scale phrase-based system described in Koehn et al. (2003) as a baseline. The baseline includes two bi-directional phrase-based models ($P_{count}(\tilde{e}|\ \tilde{f})$ and $P_{count}(\tilde{f}|\ \tilde{e})$), three penalties for word, phrase and distortion, and finally, the language model. On top of the baseline, we build four different systems, each augmented with a domain-invariance feature. The first feature is the source-target coherence feature, $D(\tilde{e}, \tilde{f})$, where we use the Chebyshev distance as our default options. We also investigate the performance of other metrics including the Hellinger distance,[6] and the Kullback-Leibler divergence.[7] Our second and third features are the domain specificity of phrases on the *source* $D_\alpha(\tilde{f})$ and on the *target* $D_\alpha(\tilde{e})$ sides. Finally, we also deploy all these three domain-invariance features $D_\alpha(\tilde{f}) + D_\alpha(\tilde{e}) + D(\tilde{e}, \tilde{f}))$. The experiments are conducted for the task *Legal* on *English-German*.

English-German (Task: Legal)		
Dev	**System**	**BLEU↑**
In-domain	Baseline	28.8
	$+D(\tilde{e}, \tilde{f})$	29.1/+**0.3**
	$+D_\alpha(\tilde{e})$	29.4/+**0.6**
	$+D_\alpha(\tilde{f})$	29.8/+**1.0**
	$+D_\alpha(\tilde{f}) + D_\alpha(\tilde{e}) + D(\tilde{e}, \tilde{f})$	29.9/+**1.1**
Mixed-domains	Baseline	28.5
	$+D(\tilde{e}, \tilde{f})$	28.8/+**0.3**
	$+D_\alpha(\tilde{e})$	29.3/+**0.8**
	$+D_\alpha(\tilde{f})$	29.6/+**1.1**
	$+D_\alpha(\tilde{f}) + D_\alpha(\tilde{e}) + D(\tilde{e}, \tilde{f})$	29.8/+**1.3**
Mixed-domains (**Exclude Legal**)	Baseline	28.3
	$+D(\tilde{e}, \tilde{f})$	28.6/+**0.3**
	$+D_\alpha(\tilde{e})$	29.1/+**0.8**
	$+D_\alpha(\tilde{f})$	29.5/+**1.2**
	$+D_\alpha(\tilde{f}) + D_\alpha(\tilde{e}) + D(\tilde{e}, \tilde{f})$	29.6/+**1.3**

Table 5: Improvements over the baseline. The bold fact indicates that the difference is statistically significant.

English-German (Task: Legal)		
Dev	**Metric**	**BLEU↑**
In-domain	*Chebyshev*	29.1/+0.3
	Kullback-Leibler ($D_{KL}(\vec{\tilde{e}},\ \vec{\tilde{f}})$)	**29.2/+0.4**
	Kullback-Leibler ($D_{KL}(\vec{\tilde{f}},\ \vec{\tilde{e}})$)	29.0/+0.2
	Hellinger	29.0/+0.2

Table 6: Using different metrics as the measure of coherence.

Table 5 and Table 6 present the results. Overall, we can see that all domain-invariance features con-

[6] $D_H(\vec{\tilde{e}},\ \vec{\tilde{f}}) = \frac{1}{\sqrt{2}}\sqrt{\sum_z \left(\sqrt{P(z|\ \tilde{e})} - \sqrt{P(z|\ \tilde{f})}\right)^2}$.

[7] $D_{KL}(\vec{\tilde{e}},\ \vec{\tilde{f}}) = \sum_z P(z|\ \tilde{e}) \log \frac{P(z|\ \tilde{e})}{P(z|\ \tilde{f})}$; $D_{KL}(\vec{\tilde{f}},\ \vec{\tilde{e}}) = \sum_z P(z|\ \tilde{f}) \log \frac{P(z|\ \tilde{f})}{P(z|\ \tilde{e})}$.

German-English (Task: Legal Services)	
Input	*im jahr 2004 befindet der rat über die verpflichtung der elektronischen übertragung solcher aufzeichnungen.*
Reference	*the council shall decide in 2004 on the obligation **to transmit such records electronically**.*
Baseline	*in 2004 the council is the obligation **on the electronic transfer of such records**.*
+ $D_\alpha(\tilde{f})$	*in 2004 the council is on the obligation **of electronic transfer of such records.***
+ $D_\alpha(\tilde{e})$	*in 2004 the council is on the obligation **of electronic transmission of such records.***
+ $D(\tilde{e}, \tilde{f})$	*in 2004 the council is on the obligation **of electronic transmission of such records.***
+ ALL	*in 2004 the council is on the obligation **of electronic transmission of such records.***
Input	*die angemessenheit und wirksamkeit der internen verwaltungssysteme sowie die leistung der dienststellen*
Reference	*for assessing the suitability and effectiveness **of internal management systems** and the performance of departments*
Baseline	*the adequacy and effectiveness **of internal administrative systems** as well as the performance of the services*
+ $D_\alpha(\tilde{f})$	*the adequacy and effectiveness **of the internal management systems**, as well as the performance of the services*
+ $D_\alpha(\tilde{e})$	*the adequacy and effectiveness **of internal management systems**, as well as the performance of the services*
+ $D(\tilde{e}, \tilde{f})$	*the adequacy and effectiveness **of the internal administrative systems** as well as the performance of the services*
+ ALL	*the adequacy and effectiveness **of internal management systems**, as well as the performance of the services*
Input	*zur ausführung der ausgaben nimmt der anweisungsbefugte mittelbindungen vor, geht rechtliche verpflichtungen ein*
Reference	*to implement expenditure, the authorising officer shall **make** budget commitments and **legal commitments***
Baseline	*the implementation of expenditure, the authorising officer commitments before, **is a legal commitments***
+ $D_\alpha(\tilde{f})$	*the implementation of expenditure, the authorising officer commitments, **is a legal obligations***
+ $D_\alpha(\tilde{e})$	*the implementation of expenditure, the authorising officer commitments before, **is a legal obligations***
+ $D(\tilde{e}, \tilde{f})$	*the implementation of expenditure, the authorising officer commitments before, **is a legal commitments***
+ ALL	*the implementation of expenditure, the authorising officer commitments before, **is a legal obligations***

Table 7: Translation outputs produced by the basic **baseline** and its augmented systems with additional abstract feature functions derived from hidden domain information.

tribute to adaptation performance. Specifically, we observe the following:

- Favouring the source-target coherence across subdomains (i.e. adding the feature $D(\tilde{e}, \tilde{f})$) provides a significant translation improvement of $+0.3$ BLEU. Which specific similarity measure is used does not seem to matter that much (see Table 6). We obtain the best result (+0.4 BLEU) with the KL divergence ($D_{KL}(\vec{\tilde{e}}, \vec{\tilde{f}})$). However, the differences are not statistically significant.
- Integrating a preference for less domain-specific translation phrases at the target side ($D_\alpha(\tilde{e})$) leads to a translation improvement of $+0.6$ BLEU.
- Doing the same for the source side ($D_\alpha(\tilde{f})$), in turn, leads to an improvement of $+1.0$ BLEU.
- Augmenting the baseline by integrating all our features leads to the best result, with an improvement of $+1.1$ BLEU.
- The translation improvement is observed also for training with a development set of mixed domains (even for the **mixed-domain minus in-domain** setting when excluding the *Legal* data from the mixed development set).
- The weights for all domain-invariance features, once tuned, are positive in all the experiments.

Table 7 presents examples of translations from different systems. For example, the domain-invariant system revises the translation from "*electronic transfer*" to "*electronic transmission*" for the German phrase "*elektronischen Übertragung*", and from "*internal administrative systems*" to "*internal management systems*" for the German phrase "*internen verwaltungssysteme*". The revisions, however, are not always successful. For instance, adding $D_\alpha(\tilde{e})$ and $D_\alpha(\tilde{f})$ resulted in revising the translation of the German phrase "*rechtliche verpflichtungen*" to "*legal obligations*", which is a worse choice (at least according to BLEU) than "*legal commitments*" produced by the baseline.

We also present a brief analysis of latent subdomains induced by our projection frame-

English-German													
Task	**Baseline**	**Our System**											
		+z$_1$	**+z$_2$**	**+z$_3$**	**+z$_4$**	**+z$_5$**	**+z$_6$**	**+z$_7$**	**+z$_8$**	**+z$_9$**	**+z$_{10}$**	**+z$_{11}$**	**+z$_{12}$**
Hardware	20.2	20.4	20.4	20.4	**20.5**	**20.5**	**20.5**	20.4	20.4	**20.5**	20.4	20.4	20.4
Software	*22.8*	23.0	23.0	23.0	*22.8*	*22.9*	**23.1**	23.0	23.0	23.0	23.0	23.0	*22.8*
P&B Services	13.3	**13.6**	**13.6**	*13.3*	13.5	**13.6**	**13.6**	13.5	13.5	**13.6**	13.5	**13.6**	13.5
Legal	28.5	28.7	*28.6*	**29.1**	28.7	*28.6*	28.9	28.8	28.8	28.9	*28.6*	*28.6*	28.8

Table 8: Latent Subdomain Analysis (with BLEU score).

work. For each subdomain z we integrate the domain posteriors ($P(z|\ \tilde{e})$ and $P(z|\ \tilde{f})$ and the source-target domain-coherence feature $\left|P(z|\ \tilde{e}) - P(z|\ \tilde{f})\right|$). We hypothesize that whenever we observe an improvement for a translation task with domain-informed features, this means that the corresponding latent subdomain z is *close* to the target translation domain.

The results are presented in Table 8. Apparently, among the latent subdomains, z_4, z_5, z_6, z_9 are closest to the target domain of *Hardware*. Their derived feature functions are helpful in improving the translation accuracy for the task. Similarly, z_1, z_2, z_5, z_6, z_9 and z_{11} are closest to *Professional & Business*, z_6 is closest to *Software*, and z_3 is closest to *Legal*. Meanwhile, z_4, z_5 and z_{12} are not relevant to the task of *Software*. Similarly, z_3 is not relevant to *Professional & Business*, and z_2, z_5 and z_{10} are not relevant to *Legal*.

Using topic models instead of latent domains. Our domain-invariance framework demands access to posterior distributions of latent domains for phrases. Though we argued for using our domain induction approach, other latent variable models can be used to compute these posteriors. One natural option is to use topic models, and more specifically LDA (Blei et al., 2003). Will our domain-invariance framework still work with topic models, and how closely related are the induced latent domains induced with LDA and our model? These are the questions we study in this section.

We estimate LDA at the sentence level in a monolingual regime[8] on one side of each parallel corpus (let us assume for now that this is the source side). When the model is estimated, we obtain the posterior distributions of topics (we denote them as z, as we treat them as domains) for each source-side sentence in the training set. Now, as we did with our phrase induction framework, we associate these posteriors with every phrase both in the source and in the target sides of that sentence pair. Phrase and phrase-pair features defined in Section 2 are computed relying on these probabilities averaged over the entire training set. We try both directions, that is also estimating LDA on the target side and transferring the posterior probabilities to the source side.

In order to estimate LDA, we used Gibbs sampling implemented in the Mallet package (McCallum, 2002) with default values of hyper-parameters ($\alpha = 0.01$ and $\beta = 0.01$). Table 9 presents the results for the Legal task with three different system optimization settings. BLEU, METEOR and TER are reported. As the result suggests, using our induction framework tends to yield slightly better translation results in terms of METEOR and especially BLEU. However, using LDA seems to lead to slightly better translation result in terms of TER.

English-German (Task: Legal)				
Dev	**Algorithms**	**BLEU↑**	**METEOR↑**	**TER↓**
In-domain	Our	29.9	33.1	55.5
	LDA (source)	29.9	33.1	55.4
	LDA (target)	29.9	33.1	**55.3**
Mixed-domains	Our	**29.8**	32.9	54.9
	LDA (source)	29.7	32.9	**54.8**
	LDA (target)	29.7	32.9	**54.8**
Mixed-domains (**Exclude Legal**)	Our	**29.6**	**32.8**	54.6
	LDA (source)	29.4	32.7	**54.5**
	LDA (target)	29.4	32.7	54.6

Table 9: Comparison in latent domain induction with various algorithms.

Topics in LDA-like models encode co-occurrence patterns in bag-of-word representations of sentences. In contrast, domains in our domain-induction framework rely on ngrams and word-

[8]Note that bilingual LDA models (e.g., see (Hasler et al., 2014; Zhang et al., 2014)) could potentially produce better results but we leave them for future work.

alignment information. Consequently, these models are likely to encode different latent information about sentences. We also investigate translation performance when we use both coherence features from LDA and coherence features from our own framework. Table 10 shows that using all the induced coherence features results in the best translation, no matter which translation metric is used. We leave the exploration of such an extension for future work.

English-German (Task: Legal)

Dev	Algorithms	BLEU↑	METEOR↑	TER↓
Mixed domains	Our features	**29.8**	32.9	54.9
	LDA (source) features	29.7	32.9	54.8
	All Features	**29.8**	**33.0**	**54.7**

Table 10: Combination of all features.

5 Related Work and Discussion

Domain adaptation is an important challenge for many NLP problems. A good survey of potential translation errors in MT adaptation can be found in Irvine et al (2013). Lexical selection appears to be the most common source of errors in domain adaptation scenarios (Irvine et al., 2013; Wees et al., 2015). Other translation errors include reordering errors (Chen et al., 2013a; Zhang et al., 2015), alignment errors (Cuong and Sima'an, 2015) and overfitting to the source domain at the parameter tuning stage (Joty et al., 2015).

Adaptation in SMT can be regarded as injecting prior knowledge about the target translation task into the learning process. Various approaches have so far been exploited in the literature. They can be loosely categorized according to the type of prior knowledge exploited for adaptation. Often, a seed in-domain corpus exemplifying the target translation task is used as a form of prior knowledge. Various techniques can then be used for adaptation. For example, one approach is to combine a system trained on the in-domain data with another general-domain system trained on the rest of the data (e.g. see (Koehn and Schroeder, 2007; Foster et al., 2010; Bisazza et al., 2011; Sennrich, 2012b; Razmara et al., 2012; Sennrich et al., 2013; Haddow, 2013; Joty et al., 2015)). Rather than using the entire training data, it is also common to combine the in-domain system with a system trained on a selected subset of the data (e.g. see (Axelrod et al., 2011; Koehn and Haddow, 2012; Duh et al., 2013; Kirchhoff and Bilmes, 2014; Cuong and Sima'an, 2014b)).

In some other cases, the prior knowledge lies in meta-information about the training data. This could be document-annotated training information (Eidelman et al., 2012; Hu et al., 2014; Hasler et al., 2014; Su et al., 2015; Zhang et al., 2014), and domain-annotated sub-corpora (Chiang et al., 2011; Sennrich, 2012b; Chen et al., 2013b; Carpuat et al., 2014; Cuong and Sima'an, 2015). Some recent approaches perform adaptation by exploiting a target domain development, or even only the source side of the development set (Sennrich, 2012a; Carpuat et al., 2013; Carpuat et al., 2014; Mansour and Ney, 2014).

Recently, there was some research on adapting simultaneously to multiple domains, the goal related to ours (Clark et al., 2012; Sennrich, 2012a). For instance, Clark et al. (2012) augment a phrase-based MT system with various domain indicator features to build a single system that performs well across a range of domains. Sennrich (2012a) proposed to cluster training data in an unsupervised fashion to build mixture models that yield good performance on multiple test domains. However, their approaches are very different from ours, that is minimizing risk associated with choosing domain-specific translations.

Moreover, the present work deviates radically from earlier work in that it explores the scenario where no prior data or knowledge is available about the translation task during training time. The focus of our approach is to aim for safer translation by rewarding domain-invariance of translation rules over latent subdomains that can be (still) useful on adaptation tasks. The present study is inspired by (Zhang et al., 2014) which exploits topic-insensitivity that is learned over documents for translation. The goal and setting we are working on is markedly different (i.e. we do not have access to meta-information about the training and translation tasks at all). The domain-invariance induced is integrated into SMT systems as feature functions, redirecting the decoder to a better search space for the translation over adaptation tasks. This aims at biasing the decoder towards translations that are less domain-specific and more source-target domain coherent.

There is an interesting relation between this work and extensive prior work on minimum Bayes risk (MBR) objectives (used either at test time (Kumar and Byrne, 2004) or during training (Smith and Eisner, 2006; Pauls et al., 2009)). As with our work, the goal of MBR minimization is to select translations that are less "risky". Their risk is due to the uncertainty in model predictions, and some of this uncertainty may indeed be associated with domain-variability of translations. Still, a system trained with an MBR objective will tend to output most frequent translation rather than the most domain-invariant one, and this, as we argued in the introduction, might not be the right decision when applying it across domains. We believe that the two classes of methods are largely complementary, and leave further investigation for future work.

At a conceptual level it is also related to regularizers used in learning domain-invariant neural models (Titov, 2011), specifically autoencoders. Though they also consider divergences between distributions of latent variable vectors, they use these divergences at learning time to bias models to induce representations maximally invariant across domains. Moreover, they assume access to meta-information about domains and consider only classification problems.

6 Conclusion

This paper aims at adapting machine translation systems to all domains at once by favoring phrases that are domain-invariant, that are safe to use across a variety of domains. While typical domain adaptation systems expect a sample of the target domain, our approach does not require one and is directly applicable to any domain adaptation scenario. Experiments show that the proposed approach results in modest but consistent improvements in BLEU, METEOR and TER. To the best of our knowledge, our results are the first to suggest consistent and significant improvement by a fully unsupervised adaptation method across a wide variety of translation tasks.

The proposed adaptation framework is fairly simple, leaving much space for future research. One potential direction is the introduction of additional features relying on the assignment of phrases to domains. The framework for inducing latent domains proposed in this paper should be beneficial in this future work. The implementation of our subdomain-induction framework is available at `https://github.com/hoangcuong2011/UDIT`.

Acknowledgements

We thank anonymous reviewers for their constructive comments on earlier versions. We also thank Hui Zhang for his help on expected Kneser-Ney smoothing technique. The first author is supported by the EXPERT (EXPloiting Empirical appRoaches to Translation) Initial Training Network (ITN) of the European Union's Seventh Framework Programme. The second author is supported by VICI grant nr. 277-89-002 from the Netherlands Organization for Scientific Research (NWO). We thank TAUS for providing us with suitable data.

References

Amittai Axelrod, Xiaodong He, and Jianfeng Gao. 2011. Domain adaptation via pseudo in-domain data selection. In *Proceedings of EMNLP*.

Arianna Bisazza, Nick Ruiz, and Marcello Federico. 2011. Fill-up versus interpolation methods for phrase-based smt adaptation. In *IWSLT*.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *JMLR*.

Peter F. Brown, Vincent J. Della Pietra, Stephen A. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical machine translation: parameter estimation. *Comput. Linguist.*

Marine Carpuat, Hal Daume III, Katharine Henry, Ann Irvine, Jagadeesh Jagarlamudi, and Rachel Rudinger. 2013. Sensespotting: Never let your parallel data tie you to an old domain. In *Proceedings of ACL*.

Marine Carpuat, Cyril Goutte, and George Foster. 2014. Linear mixture models for robust machine translation. In *Proc. of WMT*.

Boxing Chen, George Foster, and Roland Kuhn. 2013a. Adaptation of reordering models for statistical machine translation. In *Proceedings of NAACL*.

Boxing Chen, Roland Kuhn, and George Foster. 2013b. Vector space model for adaptation in statistical machine translation. In *Proceedings of the ACL*.

Colin Cherry and George Foster. 2012. Batch tuning strategies for statistical machine translation. In *Proceedings of the NAACL-HLT*.

David Chiang, Steve DeNeefe, and Michael Pust. 2011. Two easy improvements to lexical weighting. In *Proceedings of ACL (Short Papers)*.

Jonathan Clark, Chris Dyer, Alon Lavie, and Noah A. Smith. 2011. Better hypothesis testing for statistical machine translation: Controlling for optimizer instability. In *Proceedings of ACL (Short Papers)*.

Jonathan H Clark, Alon Lavie, and Chris Dyer. 2012. One system, many domains: Open-domain statistical machine translation via feature augmentation. *Conference of the Association for Machine Translation in the Americas*.

Trevor Cohn and Gholamreza Haffari. 2013. An infinite hierarchical bayesian model of phrasal translation. In *Proceedings of the ACL*.

Hoang Cuong and Khalil Sima'an. 2014a. Latent domain phrase-based models for adaptation. In *Proceedings of EMNLP*.

Hoang Cuong and Khalil Sima'an. 2014b. Latent domain translation models in mix-of-domains haystack. In *Proceedings of COLING*.

Hoang Cuong and Khalil Sima'an. 2015. Latent domain word alignment for heterogeneous corpora. In *Proceedings of NAACL-HLT*.

A. P. Dempster, N. M. Laird, and D. B. Rubin. 1977. Maximum likelihood from incomplete data via the em algorithm. *JRSS, SERIES B*, 39(1):1–38.

John DeNero, Dan Gillick, James Zhang, and Dan Klein. 2006. Why generative phrase models underperform surface heuristics. In *Proc. of WMT*.

Michael Denkowski and Alon Lavie. 2011. Meteor 1.3: Automatic metric for reliable optimization and evaluation of machine translation systems. In *Proc. of WMT*.

Kevin Duh, Graham Neubig, Katsuhito Sudoh, and Hajime Tsukada. 2013. Adaptation data selection using neural language models: Experiments in machine translation. In *Proceedings of the ACL*.

Vladimir Eidelman, Jordan Boyd-Graber, and Philip Resnik. 2012. Topic models for dynamic translation model adaptation. In *ACL (Short Papers)*.

George Foster, Cyril Goutte, and Roland Kuhn. 2010. Discriminative instance weighting for domain adaptation in statistical machine translation. In *Proceedings of EMNLP*.

Michel Galley and Christopher D. Manning. 2008. A simple and effective hierarchical phrase reordering model. In *Proceedings of EMNLP*.

Kuzman Ganchev, Ben Taskar, Fernando Pereira, and Jo ao Gama. 2009. Posterior vs parameter sparsity in latent variable models. In *Proceedings of NIPS*.

Barry Haddow. 2013. Applying pairwise ranked optimisation to improve the interpolation of translation models. In *Proceedings of NAACL-HLT*.

Eva Hasler, Phil Blunsom, Philipp Koehn, and Barry Haddow. 2014. Dynamic topic adaptation for phrase-based mt. In *Proceedings of EACL*.

Kenneth Heafield, Ivan Pouzyrevsky, Jonathan H. Clark, and Philipp Koehn. 2013. Scalable modified kneser-ney language model estimation. In *Proceedings of the ACL (Volume 2: Short Papers)*.

Yuening Hu, Ke Zhai, Vladimir Eidelman, and Jordan Boyd-Graber. 2014. Polylingual tree-based topic models for translation domain adaptation. In *Proceedings of the ACL*.

Ann Irvine, John Morgan, Marine Carpuat, Daume Hal III, and Dragos Munteanu. 2013. Measuring machine translation errors in new domains. In *TACL*.

Mark Johnson. 2007. Why doesn't EM find good HMM POS-taggers? In *Proceedings of EMNLP-CoNLL*.

Shafiq Joty, Hassan Sajjad, Nadir Durrani, Kamla Al-Mannai, Ahmed Abdelali, and Stephan Vogel. 2015. How to avoid unwanted pregnancies: Domain adaptation using neural network models. In *Proceedings of EMNLP*.

Katrin Kirchhoff and Jeff Bilmes. 2014. Submodularity for data selection in machine translation. In *EMNLP*.

Philipp Koehn and Barry Haddow. 2012. Towards effective use of training data in statistical machine translation. In *Proceedings of the WMT*.

Philipp Koehn and Josh Schroeder. 2007. Experiments in domain adaptation for statistical machine translation. In *Proceedings of WMT*.

Philipp Koehn, Franz Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of NAACL*.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of ACL*.

Philipp Koehn. 2004. Statistical significance tests for machine translation evaluation. In *Proceedings of EMNLP*.

Philipp Koehn. 2005. Europarl: A Parallel Corpus for Statistical Machine Translation. In *Proceedings of MTSummit*.

Shankar Kumar and William J. Byrne. 2004. Minimum bayes-risk decoding for statistical machine translation. In *HLT-NAACL*.

Saab Mansour and Hermann Ney. 2014. Unsupervised adaptation for statistical machine translation. In *Proceedings of WMT*.

Spyros Matsoukas, Antti-Veikko I. Rosti, and Bing Zhang. 2009. Discriminative corpus weight estimation for machine translation. In *Proceedings of EMNLP*.

Andrew Kachites McCallum. 2002. Mallet: A machine learning for language toolkit. http://www.cs.umass.edu/ mccallum/mallet.

Markos Mylonakis and Khalil Sima'an. 2008. Phrase translation probabilities with itg priors and smoothing as learning objective. In *Proceedings of EMNLP*.

Graham Neubig, Taro Watanabe, Eiichiro Sumita, Shinsuke Mori, and Tatsuya Kawahara. 2011. An unsupervised model for joint phrase alignment and extraction. In *Proceedings of ACL-HLT*.

Franz Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Comput. Linguist.*, pages 19–51.

Franz Och and Hermann Ney. 2004. The alignment template approach to statistical machine translation. *Comput. Linguist.*, pages 417–449.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: A method for automatic evaluation of machine translation. In *Proceedings of ACL*.

Adam Pauls, John DeNero, and Dan Klein. 2009. Consensus training for consensus decoding in machine translation. In *Proceedings of EMNLP*.

Majid Razmara, George Foster, Baskaran Sankaran, and Anoop Sarkar. 2012. Mixing multiple translation models in statistical machine translation. In *Proceedings of ACL*.

Rico Sennrich, Holger Schwenk, and Walid Aransa. 2013. A multi-domain translation model framework for statistical machine translation. In *Proceedings of ACL*.

Rico Sennrich. 2012a. Mixture-modeling with unsupervised clusters for domain adaptation in statistical machine translation. In *Proceedings of the EAMT*.

Rico Sennrich. 2012b. Perplexity minimization for translation model domain adaptation in statistical machine translation. In *Proceedings of EACL*.

David A. Smith and Jason Eisner. 2006. Minimum risk annealing for training log-linear models. In *Proceedings of the COLING/ACL*.

Matthew Snover, Bonnie Dorr, R. Schwartz, L. Micciulla, and J. Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proceedings of AMTA*.

Jinsong Su, Deyi Xiong, Yang Liu, Xianpei Han, Hongyu Lin, Junfeng Yao, and Min Zhang. 2015. A context-aware topic model for statistical machine translation. In *Proceedings of the ACL-IJCNLP*.

Ivan Titov. 2011. Domain adaptation by constraining inter-domain variability of latent feature representation. In *Proceedings of ACL*.

Marlies van der Wees, Arianna Bisazza, Wouter Weerkamp, and Christof Monz. 2015. What's in a domain? analyzing genre and topic differences in smt. In *Proceedings of ACL-IJCNLP (short paper)*.

Hui Zhang and David Chiang. 2014. Kneser-ney smoothing on expected counts. In *Proceedings of ACL*.

Min Zhang, Xinyan Xiao, Deyi Xiong, and Qun Liu. 2014. Topic-based dissimilarity and sensitivity models for translation rule selection. *JAIR*.

Biao Zhang, Jinsong Su, Deyi Xiong, Hong Duan, and Junfeng Yao. 2015. Discriminative reordering model adaptation via structural learning. In *IJCAI*.