Learning an Expert from Human Annotations in Statistical Machine Translation: the Case of Out-of-Vocabulary Words

Wilker Aziz[*], **Marc Dymetman**[†], **Shachar Mirkin**[§], **Lucia Specia**[‡], **Nicola Cancedda**[†], **Ido Dagan**[§]
[*]University of São Paulo, [†]Xerox Research Centre Europe
[§]Bar-Ilan University, [‡]University of Wolverhampton
will.aziz@gmail.com, {dymetman,cancedda}@xrce.xerox.com
L.Specia@wlv.ac.uk, {mirkins,dagan}@cs.biu.ac.il

Abstract

We present a general method for incorporating an "expert" model into a Statistical Machine Translation (SMT) system, in order to improve its performance on a particular "area of expertise", and apply this method to the specific task of finding adequate replacements for Out-of-Vocabulary (OOV) words. Candidate replacements are paraphrases and entailed phrases, obtained using monolingual resources. These candidate replacements are transformed into "dynamic biphrases", generated at decoding time based on the context of each source sentence. Standard SMT features are enhanced with a number of new features aimed at scoring translations produced by using different replacements. Active learning is used to discriminatively train the model parameters from human assessments of the quality of translations. The learning framework yields an SMT system which is able to deal with sentences containing OOV words but also guarantees that the performance is not degraded for input sentences without OOV words. Results of experiments on English-French translation show that this method outperforms previous work addressing OOV words in terms of acceptability.

1 Introduction

When translating a new sentence, Statistical Machine Translation (SMT) systems often encounter "Out-of-Vocabulary" (OOV) words, that is, words for which no translation is provided in the system phrase table. The problem is particularly severe when bilingual data are scarce or the text to be translated is not from the same domain as the data used to train the system.

One approach consists in replacing the OOV word by a *paraphrase*, i.e. a word that is equivalent and known to the phrase-table. For instance, in the sentence "*The police hit the protester*", if the source word "*hit*" is OOV, it could be replaced by its paraphrase "*struck*". In previous work such paraphrases are learnt by "pivoting" through parallel texts involving multiple languages (Callison-Burch et al., 2006) or on the basis of monolingual data and distributional similarity metrics (Marton et al., 2009).

Mirkin et al. (2009) go beyond the use of paraphrase to incorporate the notion of an *entailed* phrase, that is, a word which is implied by the OOV word, but is not necessarily equivalent to it — for example, this could result in "*hit*" being replaced by the entailed phrase "*attacked*". Both paraphrases and entailed phrases are obtained using monolingual resources such as WordNet (Fellbaum, 1998). This approach results in higher coverage and human acceptability of the translations produced relative to approaches based only on paraphrases.

In (Mirkin et al., 2009) a replacement for the OOV word is chosen based on a score representing how well it fits the context of the input sentence, combined with the global SMT score obtained after translating multiple alternative sentences produced by alternative replacements. The combination of source and target language scores is heuristically defined as their product, and entailed phrases are only used when paraphrases are not available. This approach has several shortcomings: translating each replacement variant is wasteful and does not capitalize on the search capabilities of the decoder; the ad hoc combination of scores makes it difficult to tune the contribution of each score or to extend the approach to incorporate new features; and the enforced preference to paraphrases may result in inadequate paraphrases instead of acceptable entailed phrases.

We propose an approach which also takes into account both paraphrased and entailed words and uses a context model score, but differs from (Mirkin et al., 2009) in several crucial aspects,

mostly stemming from the fact that we *integrate* the selection of the replacement words into the SMT decoder. This has implications for both the decoding and training processes.

At *decoding* time, when translating a source sentence with an OOV word, besides the collection of biphrases[1] stored in the system phrase table, we generate a set of *dynamic biphrases* on the fly, based on the context of that specific source sentence, to address the OOV word. For example, we could derive the dynamic biphrases *(hit, a frappé)* and *(hit, a attaqué)* from the static ones *(struck, a frappé)* and *(attacked, a attaqué)*.

Such dynamic biphrases are assigned several features that characterize different aspects of the process that generated them, such as the appropriateness of the replacement in the context of the specific source sentence, allowing for example *reach* to be preferred to *strike* or *attack* in replacing *hit* in "*We hit the city at lunch time*". Dynamic and static biphrases compete during the search for an optimal translation.

At *training* time, standard techniques such as MERT (Minimum Error Rate Training) (Och, 2003), which attempt to maximize automatic metrics like BLEU (Papineni et al., 2002) based on a bilingual corpus, are directly applicable. However, as has been discussed in (Callison-Burch et al., 2006; Mirkin et al., 2009), such automatic measures are poor indicators of improvements in translation quality in presence of semantic modifications of the kind we are considering here. Therefore, we perform the training and evaluation on the basis of human annotations. We use a form of *active learning* to focus the annotation effort on a small set of candidates which are useful for the training.

Sentences containing OOV words represent a fairly small fraction of the sentences to be translated[2]. Thus, to avoid human annotation of a large sample with relatively few cases of OOV words, for the purpose of yielding a statistically unbiased sample, we perform a two-phase training: (a) the standard SMT model is first trained on an unbiased bilingual sample using MERT and N-best lists; and (b) this model is extended with additional dynamic features and iteratively updated by using other samples containing only sentences with OOV words annotated for quality by humans. We update the model parameters in such a way that the new model does not modify the scores of translations of cases without OOV words. This is done through an adaptation of the online learning algorithm MIRA (Crammer et al., 2006) which preserves linear subspaces of parameters. This approach consists therefore in learning an *expert* that is able to improve the performance of the translation system on a specific set of inputs, *while preserving its performance* on all other inputs.

The main contributions of this paper can be summarized as follows: an efficient mechanism integrated into the decoder for handling contextual information; a method for adding expertise to an SMT model relative to a specific task, relying on highly informative, biased, samples and on human scores; expert models that affect only a specific set of inputs related to a particular problem, improving the translation performance in such cases.

In the remainder of this paper, we introduce the framework proposed in this paper for learning an expert for the task of handling sentence containing OOV words (Section 2), then present our experimental setup (Section 3) and finally our results (Section 4).

2 Learning an expert for OOV words

Our approach to learning an OOV expert for SMT is motivated by several general requirements. First, for efficiency reasons, we want the expert to be tightly integrated with the SMT decoder. Second, we need to rely on human judgments of the translations produced, since automatic evaluation measures such as BLEU are poor predictors of translation quality in the presence of semantic approximations of the kind we are considering (Mirkin et al., 2009) . Third, because human annotations are costly, we need to use them sparingly. In particular: (i) we want to focus the annotation task on the specific problem of sentences containing OOV words, and (ii) even for these sentences, we should only hand the annotators a small, well-chosen, sample of translation candidates to assess, not an exhaustive list. Finally, we need to be careful not to *bias* training towards the human annotated sample in such a way that the integrated decoder becomes better on the OOV sentences, but is degraded on the "normal" sentences. We address these requirements as follows.

Integrated decoding The integrated decoder con-

[1] Biphrases are the standard source and target phrase pairs.

[2] In our experimental setting, in 50K sentences from News texts, 15% contain at least one OOV content word.

sists of a standard phrase-based SMT decoder (Lopez, 2008; Koehn, 2010) enhanced with the ability to add dynamic biphrases at runtime and attempting to maximize a variant of the standard "log-linear" objective function. The standard SMT decoder tries to find $argmax_{(a,t)}\Lambda \cdot G(s,t,a)$, where Λ is a vector of weights, and $G(s,t,a)$ a vector of features depending on the source sentence s, the target sentence t and the phrase-level alignment a. The integrated decoder tries to find

$$argmax_{(a,t)}\Lambda \cdot G(s,t,a) + M \cdot H(s,t,a)$$

where M is an additional vector of weights and $H(s,t,a)$ an additional vector of "dynamic" features associated with the dynamic biphrases and assessing different characteristics of their associated replacements (see Section 2.2). The integrated model is thus completely parametrized by the concatenated weight vector. We call this model $\Lambda \oplus M$ for short.

Human annotations We select at random a set of OOV sentences from our test domain to compose our *OOV training set*, and for each of these sentences, provide the human annotators with a sample of candidate translations for different choices of replacements. They are then asked to rank these candidates according to how well they approximate the meaning of the source. In order not to force the annotators to decide on fine-grained distinctions that they are not confident about, which could be confusing and increase noise for the learning module, we provide guidelines and an annotation interface that encourage ranking the candidates in a few distinct "clusters", where the rank between clusters is clear, but the elements inside each cluster are considered indistinguishable. The annotators are also asked to concentrate their judgment on the portions of the sentences which are affected by the different replacements. To cover potential cases of cognates, annotators can choose the actual OOV as the best "replacement".

Active sampling In order to keep the sample of candidate translations to be annotated for a given OOV source sentence small, but still informative for training, we adopt an *active learning* scheme (Settles, 2010; Haffari et al., 2009; Eck et al., 2005). We do not extract *a priori* a sample of translation candidates for each sentence in the OOV training set and ask the annotators to work on these samples — which would mean that they might have to compare candidates that have little chance of being selected by the end-model after training. Instead, This is an iterative process, with a slice of the OOV training set selected for each iteration. When sampling candidate translations (out of a given slice of the OOV training set) to be annotated in the next iteration, we use the translations produced by the model $\Lambda \oplus M$ obtained so far, after training on all previous samples. This guarantees that we sample the overall best candidates for each OOV sentence according to the current model. Additionally, we sample several other translations corresponding to top candidates according to individual features used in the model, including the context model score, as we will detail in Section 3. This ensures a diversity of candidates to compare, while avoiding having to ask the annotators to give feedback on candidates that do not stand a chance of being selected by the model.

Avoiding bias We train the model $\Lambda \oplus M$ aiming to guarantee that when the integrated decoder finds a new sentence containing OOV words, it will rank the translation candidates in a way consistent with the ranks that the human judges would give to these candidates; in particular it should output as its best translation a candidate that the annotators would rank in top position. However, if we tune *both* Λ and M to attain this goal, the value of Λ in the integrated decoder can differ significantly from its value in the standard decoder, say Λ_0. In that case, when decoding a non-OOV sentence, for which the dynamic features $H(s,t,a)$ are null, the integrated decoder would use Λ instead of Λ_0, possibly degrading its performance on such sentences. To avoid this problem, while training $\Lambda \oplus M$ we keep Λ fixed at the value Λ_0, in other words, we allow only M to be updated in the iterative learning process. In such a way, we preserve the original behavior of the system on standard inputs. This requires a learning technique that can be adapted in a way that the parameter vector $\Lambda \oplus M$ varies only in the linear subspace for which $\Lambda = \Lambda_0$; notice that this is different from training Λ and M separately and then learning the best mixing factor between the two models. One technique which provides a mathematically neat way to handle this requirement is MIRA (Crammer et al., 2006), an online training method in which each learning step consists in updating the current parameter vector minimally (in the sense of Euclidian distance) so

that it lies in a certain subspace determined by the current training point. It is then quite natural to add the constraint that it also lies on the subspace $\Lambda = \Lambda_0$.

2.1 Learning to rank OOV candidates with MIRA

Let us first write $\Omega \equiv \Lambda \oplus M$, and $F(s,t,a) \equiv G(s,t,a) \oplus H(s,t,a)$, and also introduce notation for the two projection operators $\pi^{(\Lambda)}(\Lambda \oplus M) = \Lambda$ and $\pi^{(M)}(\Lambda \oplus M) = M$.

Our goal when training from human annotations is that, whenever the annotators say that the translation candidate (s,t,a) is strictly better than the translation candidate (s,t',a'), then the model scores give the same result, namely are such that $\Omega \cdot F(s,t,a) > \Omega \cdot F(s,t',a')$. Our approach to learning can then be outlined as follows. Based on the value of Ω learned on previous iterations with other samples of OOV sentences, we actively sample, as previsouly described, a few candidate translations (s,t_j,a_j) for each source sentence s in the current slice of the data, and have them ranked by human annotators, preferably in a few distinct clusters. We extract at random a certain number of pairs of translation candidates $y_{j,k} \equiv ((s,t_j,a_j),(s,t_k,a_k))$, where (s,t_j,a_j) and (s,t_k,a_k) are assumed to belong to two different clusters. We then define a feature vector on candidate pairs $\Phi(y_{j,k}) \equiv F(s,t_j,a_j) - F(s,t_k,a_k)$.

The basic learning step is the following. We assume Ω to be the current value of the parameters, and $y_{j,k}$ the next pair of annotated candidates, with (without loss of generality) (s,t_j,a_j) being strictly preferred by the annotator to (s,t_k,a_k). The update from Ω to Ω' is then performed as follows:

If $\Omega.\Phi(y_{j,k}) \geq 0$ then $\Omega' := \Omega$
Else $\Omega' := \text{argmin}_{\omega} \|\omega - \Omega\|^2$ (a)
s.t. $\omega.\Phi(y_{j,k}) - \omega.\Phi(y_{k,j}) \geq 1$ (b)
and $\pi^{(\Lambda)}(\omega) = \Lambda_0$ (c)

In other words, we are learning to rank the candidates through a "pairwise comparison" approach (Li, 2009), in which whenever a candidate pair $y_{j,k}$ is ordered in opposite ways by the annotator and the model, an update of Ω is performed. This update is a simple variant of the MIRA algorithm (as presented for instance in (Crammer, 2007)), where we update the parameter Ω minimally in terms of Euclidian distance (a) such that the new parameter respects two conditions. The condition (b) forces the classification margin for the pair to become larger with the updated model than the loss currently incurred on that pair, considering that this loss is 0 when the model chooses the correct order $y_{j,k}$, and 1 when it chooses the wrong order $y_{k,j}$. The second condition (c), which is our original addition to MIRA, forces the new parameter to have an invariant Λ-projection. The solution Ω' to the constrained optimization problem above can be obtained through Lagrange multipliers (proof omitted). Assuming that we already start from a parameter Ω such that $\pi^{(\Lambda)}(\Omega) = \Lambda_0$, then the update is given by:

$$\Omega' = \Omega + \tau\, \pi^{(M)}(X),$$

where $X \equiv \Phi(y_{j,k}) - \Phi(y_{k,j}) = 2\,\Phi(y_{j,k})$ and $\tau \equiv \frac{1-\Omega.X}{\|\pi^{(M)}(X)\|^2}$.[3] As is standard with MIRA, the final value for the model is found by averaging the Ω values found by iterating the basic learning step just described.

2.2 Dynamic Features

Given an OOV word, similar to (Mirkin et al., 2009), we search for a set of candidate replacements in WordNet, considering both synonyms and hypernyms of the OOV word which are available in the biphrase table. To this set we add the OOV word itself to account for proper nouns and potential cognates. Unlike previous work, we do not explicitly give preference to any type of candidate (e.g. synonyms over hypernyms), but instead distinguish them through features associated with the new biphrases. Given a source sentence s with an OOV word (oov), we compute several feature scores for each candidate replacement (rep):

Context model score Score indicating the degree by which rep fits the context of s. Following the results reported by Mirkin et al. (2009) we apply Latent Semantic Analysis (LSA) (Deerwester et al., 1990) as the method for computing this score, using 100-dimension vectors constructed based on a corpus of the same domain as the test set. Given s and rep, we compute the cosine similarity between their LSA vectors, where the sentence's vector is the average of the vectors of all the content words in it.

[3] Technically, this ratio is only defined for $\pi^{(M)}(X) \neq 0$, i.e. for cases where the pair of translations differ in their M projections; in the rare instances where this might not be true, we can simply ignore the pair in the learning process.

Domain similarity Score representing how well rep can replace oov in general in texts of a given domain. It is computed as the cosine similarity between the LSA vectors of the two words and is intended to give preference to replacements which correspond to more frequent senses of the OOV word in that domain (McCarthy et al., 2004).

Information loss Measures the distance in WordNet's hierarchy, denoted d, between oov and rep: $S(unk, sub) = 1 - (\frac{1}{d+1})$, where the distance between synonyms is 0, and the further the hypernym is up the hierarchy, the smaller the score. This can be considered a simple approximation to the notion of *information loss*, that is, the further the rep is from the oov in a hierarchy, the fewer semantic traits exist between the two, and therefore the more information is lost if we use rep.

Identity Binary feature to mark the cases where the OOV is kept in the sentence, what we call an "identity" replacement.

Synonyms vs hypernyms Binary feature to distinguish between synonym and hypernym replacements.

Static plus dynamic Dynamic biphrases for a given source sentence can be derived from all the static biphrases containing the chosen replacement. For example, when replacing the OOV *attacked* by *accused*, a number of static biphrases having *accused* in the source side could be used to generate *(was attacked, a été accusé)*, *(he was attacked, il a été accusé)*, *(attacked, a incriminé)*, *(attacked, le)*. Although these dynamic biphrases are very different, they will be assigned the same dynamic features values. To allow for the decoder to distinguish among such biphrases, we define an additional feature as the linear combination of the feature values of the static biphrase from which the dynamic biphrase was derived.

All static features are assigned a null value in the dynamic biphrases, and all dynamic features are assigned a null value in the static biphrases.[4]

3 Experimental Setting

Data We consider the English-French translation task and a scenario where an SMT system is used to translate texts of a different domain from the one it was trained on. We train a standard phrase-based SMT system on Europarl-v4 ($\sim 1M$ sentence pairs) and use it to decode sentences from the News domain. The standard log-linear model parameters are tuned using $2K$ unseen sentences from Europarl-v4 through MERT. A 3-gram target language model is trained using $7M$ sentences of French News texts. All datasets are taken from the the WMT-09 competition[5]. For the learning framework, we take all sentences in the News Commentary domain (training, development and test sets) from WMT-09 ($\sim 75K$) and extract those containing one OOV word that is not a proper name, symbol or number ($\sim 15\%$ of the sentences). Of these, we then randomly selected $1K$ sentences for tuning the context model (*LSA tuning set*), other $1K$ sentences for tuning the SMT feature weights (*SMT tuning set*), and 500 sentences for evaluating all methods (*test set*). The data used for computing the context model and domain similarity scores is the Reuters Corpus, Volume 1 (RCV1), which is also of the News domain[6].

We experiment with the following systems:

Baseline SMT The SMT system we use, MATRAX (Simard et al., 2005), without any special treatment for OOV words, where these are simply copied to the translations.

Monolingual retrieval Method described in (Marton et al., 2009) where paraphrases for OOV words are extracted from a monolingual corpus based on similarity metrics. We use their best-performing setting with single-word paraphrases extracted from a News domain corpus with 10M sentences. The additional biphrases are statically added into the system's biphrase library and the similarity score is used as a new feature. The log-linear model is then entirely retrained with MERT and the *SMT tuning set*.

Lexical entailment Two best performing methods described in (Mirkin et al., 2009). For each sentence with an OOV word a set of alternative source sentences is generated by directly replacing each OOV word by synonyms from WordNet or – if synonyms are not found – by hypernyms. These two settings do not add features to

[4]Thus, what we have mnemonically called "dynamic features" are features that are non-null only in dynamic biphrases; some are contextual, others not.

[5]http://www.statmt.org/wmt09/.

[6]http://trec.nist.gov/data/reuters/reuters.html

the model, hence they do not require retraining:

- **SMT** All alternative source sentences are translated using a standard SMT system and the "best" translation is the one with the highest global SMT score.
- **LSA-SMT** The 20-best alternative source sentences are selected according to an LSA context model score and translated by the a standard SMT system. The "best" translation is the one that maximizes the product of the LSA and global SMT scores.

OOV expert Method proposed in this paper, as described in Section 2. The *expert* model with all dynamic features is trained on the basis of human annotations using the *SMT tuning* set. At each iteration of the learning process we sample 80 sentences for annotation by bilingual (English and French) speakers. For a given source sentence, the sampled options at each iteration consist of a random choice of 8 dynamic biphrases corresponding to different replacements, 4 additional dynamic biphrases corresponding to different ways of translating those replacements, and the top candidates according to each of our main dynamic features: *1*-best given by the information loss feature, *2*-best given by the context model feature, *1*-best given by the domain similarity feature and *1*-best given by the identity feature. In total at most 17 non-identical candidates can be produced for annotation, but typically only a dozen are found. The results reported in Section 4 are obtained after only 6 iterations.

MERT Baseline with the same settings as the *OOV expert*, but where the tuning of *all* model parameters (both static and dynamic) is done *automatically* using standard MERT with reference translations for the *SMT tuning* set, instead of our learning framework and human annotations.

4 Results

Test set Following the same guidelines used for the annotation task, two native speakers of French (and fluent speakers of English) were asked to judge translations produced by different systems on 500 source sentences, according to how well they reproduced the meaning of the source sentence. They were asked to rank the translations in a few clearly distinct clusters and to discard useless translations.

Features	μ	σ	Best	Acceptance
LID	2.477	1.465	0.4728	0.5252
ID	2.491	1.463	0.4668	0.5211
LI	2.547	1.457	0.4427	0.5050
I	2.561	1.463	0.4447	0.4970
D	2.924	1.414	0.3360	0.3722
LD	2.930	1.412	0.3340	0.3702
L	3.056	1.361	0.2857	0.3300
Baseline	3.219	1.252	0.2093	0.2918

Table 1: Comparison between different feature combinations and the baseline showing the percentage of times each combination outputs a translation that is acceptable, i.e. is not discarded (Acceptance), a translation that is ranked in the first cluster (Best), as well as the the mean rank (μ) and standard deviation (σ) of each combination, where the discarded translations are conventionally assigned a rank of 5, lower than the rank of any acceptable cluster observed among the annotations. (L) context model score, (I) information-loss, (D) domain similarity, (Baseline) SMT system.

We computed inter-annotator agreement concerning both acceptance and ranking, for translations of 30 randomly sampled source sentences that were evaluated by both annotators. For ranking, we followed (Callison-Burch et al., 2008), checking for each two translations, A and B, whether the annotators agreed that $A = B$, $A > B$ or $A < B$. This resulted in kappa coefficient scores (Cohen, 1960) of 0.87 for translation acceptance and 0.83 for ranking.

Combinations of dynamic features In order to have a picture of the contribution of each dynamic feature to the expert model, we compare the performance on the test set of different combinations of our main features. The results are shown in Table 1. The features not mentioned in the table, such as the *identity* flag, are secondary features included in all combinations.

The *baseline SMT* system (i.e., only *identity replacements*) reaches 29.18% of acceptance (a translation is said to be acceptable if it is not discarded), which is related to the fact that, for the given domain, copying an OOV English word into the French output often results in a cognate. The best performance is obtained with combination of all features (LID).

Evolution of learning For the complete feature vector LID we compared the performance (on the test data) of models corresponding to different iterations of the online learning scheme. The results are presented in Table 2. We see a large increase in performance from M_0 to M_1, then smaller increases. After two or three iterations the perfor-

Iterations	μ	σ	Best	Acceptance
M_6	2.487	1.458	0.4628	0.5252
M_5	2.491	1.459	0.4628	0.5231
M_4	2.489	1.458	0.4628	0.5252
M_3	2.493	1.455	0.4588	0.5252
M_2	2.501	1.456	0.4567	0.5211
M_1	2.519	1.456	0.4507	0.5151
M_0	2.944	1.407	0.328	0.3642
Baseline	3.237	1.228	0.1932	0.2918

Table 2: Each iteration adds 80 annotated sentences to the training set, from which the next vector of weights is computed. The dynamic vector M_0 was initialized with zero for the replacement-related features and 1 for the source-target feature. (Baseline) SMT system without OOV handling.

mance changes are negligible, indicating that annotation effort for training the system could be roughly divided by two without affecting its end performance.

Comparison with other systems We now compare our LID model, in different decoding and training setups, with the methods proposed in previous work and described in Section 3. Table 3 presents the results in terms of mean rank and standard deviation (note that the rank is relative to the other systems in the comparison and is not directly comparable to the rank of the same system in a different comparison), percentage of time each system outputs a first-ranked translation and the percentage of time it outputs an acceptable one, using the same conventions as in Table 1.

Let us first focus on the lines other than b-LID in the table, corresponding to systems mentioned in Section 3. These results are consistent across different measures: acceptance, mean rank, or being ranked in the best cluster. In particular we see that both the LID, trained on human annotations, and LID-MERT systems, trained by MERT from reference translations, considerably outperform the baseline and the Monolingual Retrieval method, with LID being better than LID-MERT particularly in terms of acceptability. A somewhat disappointing result, however, is that LID is inferior to both SMT-LSA and SMT on all measures.

By observing the outputs of SMT and SMT-LSA, we noticed that, although they can theoretically produce identity replacements, they never actually do so on the test set. This is probably due to the fact that the language model that is part of the scoring function in both SMT and SMT-LSA contributes to giving a very bad score to identity replacements, unless they happen to belong to the set of possible French forms ("exact" cognates), and therefore these models tend to strongly favor entailment replacements.

On the other hand, our LID model does actually produce identity replacements quite often, some of which are acceptable (perhaps even ranked first) to the annotators, but a majority of which lead to non-acceptability. This is due to the fact that, at training time, the LID model actually learns to score the identity replacements relatively well (often overcoming the repulsion of the language model feature in the underlying baseline SMT system), due to the fact that many of them are actually preferred by the annotators, typically those that correspond to approximate cognates of existing French words (the annotation guidelines did not discourage them from doing so). Thus the LID model has a tendency to sometimes favor identities over entailments. *However*, it is not clever enough to distinguish the "good" identities (namely, the quasi-cognates) from the bad ones (namely, English words with no obvious French connotation), given that all identity replacements are only identified by a binary feature (identity vs. non-identity) instead of being associated with any features that could predict their understandability in a French sentence. Thus LID, when it selects an identity replacement, often selects an unacceptable one.

Motivated by this uncertainty concerning the use of identity replacements, we defined a system b-LID which uses the same model as LID, but the identity replacements are blocked at *decoding time*. In this way the system is forced to produce an entailment replacement instead of an identity one, but otherwise ranks the different entailment replacements in the same order as the original LID. We can then see from Table 3 that b-LID outperforms every other system by a large margin:[7] it is excellent at distinguishing between true entailments, and while it misses some good identity replacements, is not handicapped in this respect relative to the other systems, which are also unable to model them.

5 Conclusions

While our approach is motivated by a specific problem (OOV terms), we believe that some of the innovations we have introduced are of a larger

[7] A Wilcoxon signed rank test (Wilcoxon, 1945) shows that b-LID is better ranked than its closest competitor SMT with a p-value of less than 2%.

System	μ	σ	Best	Acceptance
b-LID	2.274	1.803	0.6258	0.7002
SMT	2.736	1.933	0.5172	0.5822
SMT-LSA	2.744	1.931	0.5132	0.5822
LID	3.018	1.913	0.4145	0.5252
LID-$mert$	3.153	1.928	0.4024	0.4849
Baseline	3.998	1.603	0.1549	0.2918
MonRet	4.107	1.584	0.1690	0.2495

Table 3: (LID) complete dynamic vector trained on the basis of human assessments; (b-LID) as LID, but blocking identity replacements; (LID-MERT) complete dynamic vector trained on the basis of automatic assessments; (SMT, SMT-LSA) and (MonRet or Monolingual retrieval) as described in Section 3; (Baseline) SMT system without OOV handling.

general interest for SMT: our use of dynamic biphrases and features for incorporating complex additional run-time knowledge into a standard phrase-based SMT system, our approach to integrating a MERT-trained log-linear model with a model actively trained from a small sample of human annotations addressing a specific phenomenon, and finally the formal techniques used in order to guarantee that the expert that is thus learned from a focussed, biased, sample, is able to improve performance on its domain of expertise while preserving the baseline system's performance on the standard cases.

Acknowledgments

This work was supported in part by the ICT Programme of the European Community, under the PASCAL-2 Network of Excellence, ICT-216886. We thank Binyam Gebrekidan Gebre and Ibrahim Soumana for performing the annotations and the anonymous reviewers for their useful comments. This publication only reflects the authors' views.

References

Chris Callison-Burch, Philipp Koehn, and Miles Osborne. 2006. Improved Statistical Machine Translation Using Paraphrases. In *Proceedings of HLT-NAACL*.

Chris Callison-Burch, Cameron Fordyce, Philipp Koehn, Christof Monz, and Josh Schroeder. 2008. Further Meta-Evaluation of Machine Translation. In *Proceedings of WMT*.

Jacob Cohen. 1960. A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, 20(1):37–46.

Koby Crammer, Ofer Dekel, Joseph Keshet, Shai Shalev-Shwartz, and Yoram Singer. 2006. Online passive-aggressive algorithms. *Journal of Machine Learning Research*, 7:551–585.

Koby Crammer. 2007. Online learning of real-world problems. Tutorial given at ICML. www.cis.upenn.edu/~crammer/icml-tutorial-index.html.

Scott Deerwester, S.T. Dumais, G.W. Furnas, T.K. Landauer, and R.A. Harshman. 1990. Indexing by Latent Semantic Analysis. *Journal of the American Society for Information Science, 41*, pages 391–407.

Matthias Eck, Stephan Vogel, and Alex Waibel. 2005. Low cost portability for statistical machine translation based on n-gram coverage. In *MT Summit X*, pages 227–234.

Christiane Fellbaum, editor. 1998. *WordNet: An Electronic Lexical Database (Language, Speech, and Communication)*. The MIT Press.

Gholamreza Haffari, Maxim Roy, and Anoop Sarkar. 2009. Active learning for statistical phrase-based machine translation. In *Proceedings of Human Language Technologies / North American Chapter of the Association for Computational Linguistics*, pages 415–423.

Philipp Koehn. 2010. *Statistical Machine Translation*. Cambridge University Press.

Hang Li. 2009. Learning to rank. Tutorial given at ACL-IJCNLP, August. research.microsoft.com/en-us/people/hangli/li-acl-ijcnlp-2009-tutorial.pdf.

Adam Lopez. 2008. Statistical machine translation. *ACM Computing Surveys*, 40(3):1–49.

Yuval Marton, Chris Callison-Burch, and Philip Resnik. 2009. Improved statistical machine translation using monolingually-derived paraphrases. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 381–390.

Diana McCarthy, Rob Koeling, Julie Weeds, and John Carroll. 2004. Finding predominant word senses in untagged text. In *In Proceedings of ACL*, pages 280–287.

Shachar Mirkin, Lucia Specia, Nicola Cancedda, Ido Dagan, Marc Dymetman, and Idan Szpektor. 2009. Source-language entailment modeling for translating unknown terms. In *Proceedings of ACL*, Singapore.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *ACL '03: Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, pages 160–167.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of ACL*.

Burr Settles. 2010. Active learning literature survey. Technical report, University of Wisconsin-Madison.

M. Simard, N. Cancedda, B. Cavestro, M. Dymetman, E. Gaussier, C. Goutte, and K. Yamada. 2005. Translating with Non-contiguous Phrases. In *Proceedings of HLT-EMNLP*.

Frank Wilcoxon. 1945. Individual Comparisons by Ranking Methods. *Biometrics Bulletin*, 1(6):80–83.