

Both the first and the second terms are sense-dependent, and each factors as,

$$P(y|x^e, x^f, z=k; \theta) \propto \Psi(x^e, z=k, y) \Psi(x^f, y) \\ = \exp(\mathbf{y}^T \mathbf{x}_k^e) \exp(\mathbf{y}^T \mathbf{x}^f) = \exp(\mathbf{y}^T (\mathbf{x}_k^e + \mathbf{x}^f)),$$

where \mathbf{x}_k^e is the embedding corresponding to the k^{th} sense of the word x^e , and y is either y^e or y^f . The factor $\Psi(x^e, z=k, y)$ use the corresponding sense vector in a skip-gram-like formulation. This results in total of 4 factors,

$$P(y^e, y^f | z, x^e, x^f; \theta) \propto \Psi(x^e, z, y^e) \Psi(x^f, y^f) \\ \Psi(x^e, z, y^f) \Psi(x^f, y^e) \quad (4)$$

See Figure 2 for illustration of each factor. This modeling approach is reminiscent of (Luong et al., 2015), who jointly learned embeddings for two languages l_1 and l_2 by optimizing a joint objective containing 4 skip-gram terms using the aligned pair (x^e, x^f) —two predicting monolingual contexts $l_1 \rightarrow l_1, l_2 \rightarrow l_2$, and two predicting crosslingual contexts $l_1 \rightarrow l_2, l_2 \rightarrow l_1$.

Learning. Learning involves maximizing the log-likelihood,

$$P(y^e, y^f | x^e, x^f; \alpha, \theta) = \int_{\beta} \sum_z P(y^e, y^f, z, \beta | x^e, x^f, \alpha; \theta) d\beta$$

for which we use variational approximation. Let $q(z, \beta) = q(z)q(\beta)$ where

$$q(z) = \prod_i q(z_i) \quad q(\beta) = \prod_{w=1}^V \prod_{k=1}^T \beta_{wk} \quad (5)$$

are the fully factorized variational approximation of the true posterior $P(z, \beta | y^e, y^f, x^e, x^f, \alpha)$, where V is the size of english vocabulary, and T is the maximum number of senses for any word. The optimization problem solves for $\theta, q(z)$ and $q(\beta)$ using the stochastic variational inference technique (Hoffman et al., 2013) similar to (Bartunov et al., 2016) (refer for details).

The resulting learning algorithm is shown as Algorithm 1. The first for-loop (line 1) updates the English sense vectors using the crosslingual and monolingual contexts. First, the expected sense distribution for the current English word w is computed using the current estimate of $q(\beta)$ (line 4). The sense distribution is updated (line 7) using the combined monolingual and crosslingual contexts

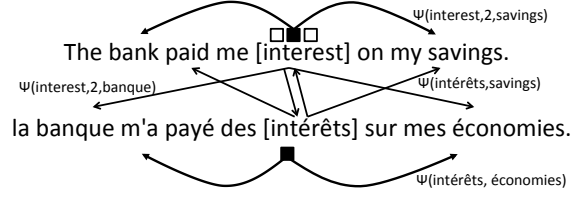


Figure 2: The aligned pair $(interest, intérêt)$ is used to predict monolingual and crosslingual context in both languages (see factors in eqn. (4)). We pick each sense (here 2nd) vector for *interest*, to perform weighted update. We only model polysemy in English.

(line 5) and re-normalized (line 8). Using the updated sense distribution $q(\beta)$'s sufficient statistics is re-computed (line 9) and the global parameter θ is updated (line 10) as follows,

$$\theta \leftarrow \theta + \rho_t \nabla_{\theta} \sum_{k|z_{ik} > \epsilon} \sum_{y \in y_c} z_{ik} \log p(y|x_i, k, \theta) \quad (6)$$

Note that in the above sum, a sense participates in a update only if its probability exceeds a threshold ϵ ($= 0.001$). The final model retains sense vectors whose sense probability exceeds the same threshold. The last for-loop (line 11) jointly optimizes the foreign embeddings using English context with the standard skip-gram updates.

Disambiguation. Similar to (Bartunov et al., 2016), we can disambiguate the sense for the word x^e given a monolingual context y^e as follows,

$$P(z | x^e, y^e) \propto P(y^e | x^e, z; \theta) \sum_{\beta} P(z | x^e, \beta) q(\beta) \quad (7)$$

Although the model trains embeddings using both monolingual and crosslingual context, we only use monolingual context at test time. We found that so long as the model has been trained with multilingual context, it performs well in sense disambiguation on new data even if it contains only monolingual context. A similar observation was made by (Šuster et al., 2016).

4 Multilingual Extension

Bilingual distributional signal alone may not be sufficient as polysemy may survive translation in the second language. Unlike existing approaches, we can easily incorporate multilingual distributional signals in our model. For using languages l_1 and l_2 to learn multi-sense embeddings for English, we train on a concatenation of En- l_1 parallel corpus with an En- l_2 parallel corpus. This technique can easily be generalized to more than

Algorithm 1 Psuedocode of Learning Algorithm

Input: parallel corpus $E = \{x_1^e, \dots, x_{N_e}^e\}$ and $F = \{x_1^f, \dots, x_{N_f}^f\}$ and alignments $A_{e \rightarrow f}$ and $A_{f \rightarrow e}$, Hyper-parameters α and T , window sizes d, d' .

Output: $\theta, q(\beta), q(\mathbf{z})$

- 1: **for** $i = 1$ to N_e **do** \triangleright update english vectors
- 2: $w \leftarrow x_i^e$
- 3: **for** $k = 1$ to T **do**
- 4: $z_{ik} \leftarrow \mathbb{E}_{q(\beta_w)}[\log p(z_i = k | x_i^e)]$
- 5: $y_c \leftarrow \text{Nbr}(x_i^e, E, d) \cup \text{Nbr}(x_i^f, F, d') \cup \{x_i^f\}$
 where $x_i^f = A_{e \rightarrow f}(x_i^e)$
- 6: **for** y in y_c **do**
- 7: SENSE-UPDATE(x_i^e, y, z_i)
- 8: Renormalize z_i using softmax
- 9: Update suff. stats. for $q(\beta)$ like (Bartunov et al., 2016)
- 10: Update θ using eq. (6)
- 11: **for** $i = 1$ to N_f **do** \triangleright jointly update foreign vectors
- 12: $y_c \leftarrow \text{Nbr}(x_i^f, F, d) \cup \text{Nbr}(x_i^e, E, d') \cup \{x_i^e\}$
 where $x_i^e = A_{f \rightarrow e}(x_i^f)$
- 13: **for** y in y_c **do**
- 14: SKIP-GRAM-UPDATE(x_i^f, y)
- 15: **procedure** SENSE-UPDATE(x_i, y, z_i)
- 16: $z_{ik} \leftarrow z_{ik} + \log p(y | x_i, k, \theta)$

two foreign languages to obtain a large multilingual corpus.

Value of $\Psi(y^e, x^f)$. The factor modeling the dependence of the English context word y^e on foreign word x^f is crucial to performance when using multiple languages. Consider the case of using French and Spanish contexts to disambiguate the financial sense of the English word *bank*. In this case, the (financial) sense vector of *bank* will be used to predict vector of *banco* (Spanish context) and *banque* (French context). If vectors for *banco* and *banque* do not reside in the same space or are not close, the model will incorrectly assume they are different contexts to introduce a new sense for *bank*. This is precisely why the bilingual models, like that of (Šuster et al., 2016), cannot be extended to multilingual setting, as they pre-train the embeddings of second language before running the multi-sense embedding process. As a result of naive pre-training, the French and Spanish vectors of semantically similar pairs like (*banco, banque*) will lie in different spaces and need not be close. A similar reason holds for (Guo et al., 2014a), as

Corpus Source	Lines (M)	EN-Words (M)
En-Fr EU proc.	≈ 10	250
En-Zh FBIS news	≈ 9.5	286
En-Es UN proc.	≈ 10	270
En-Fr UN proc.	≈ 10	260
En-Zh UN proc.	≈ 8	230
En-Ru UN proc.	≈ 10	270

Table 1: Corpus Statistics (in millions). Horizontal lines demarcate corpora from the same domain.

they use a two step approach instead of joint learning.

To avoid this, the vector for pairs like *banco* and *banque* should lie in the same space and close to each other and the sense vector for *bank*. The $\Psi(y^e, x^f)$ term attempts to ensure this by using the vector for *banco* and *banque* to predict the vector of *bank*. This way, the model brings the embedding space for Spanish and French closer by using English as a bridge language during joint training. A similar idea of using English as a bridging language was used in the models proposed in (Hermann and Blunsom, 2014) and (Coulmance et al., 2015). Beside the benefit in the multilingual case, the $\Psi(y^e, x^f)$ term improves performance in the bilingual case as well, as it forces the English and second language embeddings to remain close in space.

To show the value of $\Psi(y^e, x^f)$ factor in our experiments, we ran a variant of Algorithm 1 without the $\Psi(y^e, x^f)$ factor, by only using monolingual neighborhood $\text{Nbr}(x_i^f, F)$ in line 12 of Algorithm 1. We call this variant ONE-SIDED model and the model in Algorithm 1 the FULL model.

5 Experimental Setup

We first describe the datasets and the preprocessing methods used to prepare them. We also describe the Word Sense Induction task that we used to compare and evaluate our method.

Parallel Corpora. We use parallel corpora in English (En), French (Fr), Spanish (Es), Russian (Ru) and Chinese (Zh) in our experiments. Corpus statistics for all datasets used in our experiments are shown in Table 1. For En-Zh, we use the FBIS parallel corpus (LDC2003E14). For En-Fr, we use the first 10M lines from the Giga-EnFr corpus released as part of the WMT shared task (Callison-Burch et al., 2011). Note that the domain from which parallel corpus has been derived can affect

the final result. To understand what choice of languages provide suitable disambiguation signal, it is necessary to control for domain in all parallel corpora. To this end, we also used the En-Fr, En-Es, En-Zh and En-Ru sections of the MultiUN parallel corpus (Eisele and Chen, 2010). Word alignments were generated using `fast_align` tool (Dyer et al., 2013) in the symmetric intersection mode. Tokenization and other preprocessing were performed using `cdec`³ toolkit. Stanford Segmenter (Tseng et al., 2005) was used to preprocess the Chinese corpora.

Word Sense Induction (WSI). We evaluate our approach on word sense induction task. In this task, we are given several sentences showing usages of the same word, and are required to cluster all sentences which use the same sense (Nasirudin, 2013). The predicted clustering is then compared against a provided gold clustering. Note that WSI is a harder task than Word Sense Disambiguation (WSD)(Navigli, 2009), as unlike WSD, this task does not involve any supervision or explicit human knowledge about senses of words. We use the disambiguation approach in eq. (7) to predict the sense given the target word and four context words.

To allow for fair comparison with earlier work, we use the same benchmark datasets as (Bartunov et al., 2016) – Semeval-2007, 2010 and Wikipedia Word Sense Induction (WWSI). We report Adjusted Rand Index (ARI) (Hubert and Arabie, 1985) in the experiments, as ARI is a more strict and precise metric than F-score and V-measure.

Parameter Tuning. For fairness, we used five context words on either side to update each English word-vectors in all the experiments. In the monolingual setting, all five words are English; in the multilingual settings, we used four neighboring English words plus the one foreign word aligned to the word being updated ($d = 4$, $d' = 0$ in Algorithm 1). We also analyze effect of varying d' , the context window size in the foreign sentence on the model performance.

We tune the parameters α and T by maximizing the log-likelihood of a held out English text.⁴ The parameters were chosen from the following values $\alpha = \{0.05, 0.1, \dots, 0.25\}$, $T = \{5, 10, \dots, 30\}$. All models were trained for 10 iteration with a decay-

ing learning rate of 0.025, decayed to 0. Unless otherwise stated, all embeddings are 100 dimensional.

Under various choice of α and T , we identify only about 10-20% polysemous words in the vocabulary using monolingual training and 20-25% polysemous using multilingual training. It is evident using the non-parametric prior has led to substantially more efficient representation compared to previous methods with fixed number of senses per word.

6 Experimental Results

Setting	S-2007	S-2010	WWSI	avg. ARI	SCWS
En-Fr					
MONO	.044	.064	.112	.073	41.1
ONE-SIDED	.054	.074	.116	.081	41.9
FULL	.055	.086	.105	.082	41.8
En-Zh					
MONO	.054	.074	.073	.067	42.6
ONE-SIDED	.059	.084	.078	.074	45.0
FULL	.055	.090	.079	.075	41.7
En-FrZh					
MONO	.056	.086	.103	.082	47.3
ONE-SIDED	.067	.085	.113	.088	44.6
FULL	.065	.094	.120	.093	41.9

Table 2: Results on word sense induction (left four columns) in ARI and contextual word similarity (last column) in percent correlation. Language pairs are separated by horizontal lines. Best results shown in **bold**.

We performed extensive experiments to evaluate the benefit of leveraging bilingual and multilingual information during training. We also analyze how the different choices of language family (i.e. using more distant vs more similar languages) affect performance of the embeddings.

6.1 Word Sense Induction Results.

The results for WSI are shown in Table 2. Recall that the ONE-SIDED model is the variant of Algorithm 1 without the $\Psi(y^e, x^f)$ factor. MONO refers to the AdaGram model of (Bartunov et al., 2016) trained on the English side of the parallel corpus. In all cases, the MONO model is outperformed by ONE-SIDED and FULL models, showing the benefit of using crosslingual signal in training. Best performance is attained by the multilingual model (En-FrZh), showing value of multilingual signal. The value of $\Psi(y^e, x^f)$ term is also verified by the fact that the ONE-SIDED model performs worse than the FULL model.

³github.com/redpony/cdec

⁴first 100k lines from the En-Fr Europarl (Koehn, 2005)