Integrating Weakly Supervised Word Sense Disambiguation into Neural Machine Translation

Xiao Pu[*]
Nuance Communications
xiao.pu@nuance.com

Nikolaos Pappas
Idiap Research Institute
nikolaos.pappas@idiap.ch

James Henderson
Idiap Research Institute
james.henderson@idiap.ch

Andrei Popescu-Belis
HEIG-VD / HES-SO
andrei.popescu-belis@heig-vd.ch

Abstract

This paper demonstrates that word sense disambiguation (WSD) can improve neural machine translation (NMT) by widening the source context considered when modeling the senses of potentially ambiguous words. We first introduce three adaptive clustering algorithms for WSD, based on k-means, Chinese restaurant processes, and random walks, which are then applied to large word contexts represented in a low-rank space and evaluated on SemEval shared-task data. We then learn word vectors jointly with sense vectors defined by our best WSD method, within a state-of-the-art NMT system. We show that the concatenation of these vectors, and the use of a sense selection mechanism based on the weighted average of sense vectors, outperforms several baselines including sense-aware ones. This is demonstrated by translation on five language pairs. The improvements are more than 1 BLEU point over strong NMT baselines, +4% accuracy over all ambiguous nouns and verbs, or +20% when scored manually over several challenging words.

1 Introduction

The correct translation of polysemous words remains a challenge for machine translation (MT). Although some translation options may be interchangeable, substantially different senses of source words must generally be rendered by different words in the target language. Hence, an MT system should identify—implicitly or explicitly—the correct sense conveyed by each occurrence in order to generate an appropriate translation. For instance, in the following sentence from Europarl, the translation of "deal" should convey the sense "to handle" (in French *traiter*) and not "to cope" (in French *remédier*, which is wrong):

Source: How can we guarantee the system of prior notification for high-risk products at ports that have the necessary facilities to *deal* with them?

Reference translation: Comment pouvons-nous garantir le système de notification préalable pour les produits présentant un risque élevé dans les ports qui disposent des installations nécessaires pour *traiter* ces produits ?

Baseline neural MT: [. . .] les ports qui disposent des moyens nécessaires pour y *remédier* ?

Sense-aware neural MT: [. . .] les ports qui disposent des installations nécessaires pour les *traiter* ?

Current MT systems perform word sense disambiguation implicitly, based on co-occurring words in a rather limited context. In phrase-based statistical MT, the context size is related to the order of the language model (often between 3 and 5) and to the length of n-grams in the phrase table (seldom above 5). In attention-based neural MT (NMT), the context extends to the entire sentence, but

[*] Work conducted while at the Idiap Research Institute.

Transactions of the Association for Computational Linguistics, vol. 6, pp. 635–650, 2018. Action Editor: Eneko Agirre.
Submission batch: 8/2018; Published 12/2018.

multiple word senses are not modeled explicitly. The implicit sense information captured by word representations used in NMT leads to a bias in the attention mechanism towards dominant senses. Therefore, the NMT decoders cannot clearly identify the contexts in which one word sense should be used rather than another one. Hence, although NMT can use local constraints to translate "great rock band" into French as *superbe groupe de rock* rather than *grande bande de pierre*—thus correctly assigning the musical rather than geological sense to "rock"—it fails to do so for word senses that require larger contexts.

In this paper, we demonstrate that the explicit modeling of word senses can be helpful to NMT by using combined vector representations of word types and senses, which are inferred from contexts that are larger than that of state-of-the-art NMT systems. We make the following contributions:

- Weakly supervised word sense disambiguation (WSD) approaches integrated into NMT, based on three adaptive clustering methods and operating on large word contexts.
- Three sense selection mechanisms for integrating WSD into NMT, respectively based on top, average, and weighted average (i.e., attention) of word senses.
- Consistent improvements against baseline NMT on five language pairs: from English (EN) into Chinese (ZH), Dutch (NL), French (FR), German (DE), and Spanish (ES).

The paper is organized as follows. In §2, we present three adaptive WSD methods based on k-means clustering, the Chinese restaurant process, and random walks. In §3, we present three sense selection mechanisms that integrate the word senses into NMT. The experimental details appear in §4, and the results concerning the optimal parameter settings are presented in §5, where we also show that our WSD component is competitive on the SemEval 2010 shared task. §6 presents our results: The BLEU scores increase by about 1 point with respect to a strong NMT baseline, and the accuracy of ambiguous noun and verb translation improves by about 4%, while a manual evaluation of several challenging and frequent words shows an improvement of about 20%. A discussion of related work appears finally in §7.

2 Adaptive Sense Clustering for MT

In this section, we present the three unsupervised or weakly supervised WSD methods used in our experiments, which aim at clustering different occurrences of the same word type according to their senses. We first consider all nouns and verbs in the source texts that have more than one sense in WordNet, and extract from there the definition of each sense and, if available, the example. For each occurrence of such nouns or verbs in the training data, we use word2vec to build word vectors for their contexts (i.e., neighboring words). All vectors are passed to an unsupervised clustering algorithm, possibly instantiated with WordNet definitions or examples. The resulting clusters can be numbered and used as labels, or their centroid word vector can be used as well, as explained in §3.

This approach answers several limitations of previous supervised or unsupervised WSD methods. On the one hand, supervised methods require data with manually sense-annotated labels and are thus limited to typically small subsets of all word types—for example, up to one hundred content words targeted in SemEval 2010[1] (Manandhar et al., 2010) and up to a thousand words in SemEval 2015 (Moro and Navigli, 2015). In contrast, our method does not require labeled texts for training, and applies to all word types with multiple senses in WordNet (e.g., nearly 4,000 for some data sets; see Table 1 later in this paper). On the other hand, unsupervised methods often predefine the number of possible senses for all ambiguous words before clustering their occurrences, and do not adapt to what is actually observed in the data; as a result, the senses are often too fine-grained for the needs of MT, especially for a particular domain. In contrast, our model learns the number of senses for each analyzed ambiguous word directly from the data.

2.1 Definitions and Notations

For each noun or verb type W_t appearing in the training data, as identified by the Stanford POS tagger,[2] we extract the senses associated to it in WordNet[3] (Fellbaum, 1998) using NLTK.[4]

[1] www.cs.york.ac.uk/semeval2010_WSI.
[2] nlp.stanford.edu/software.
[3] wordnet.princeton.edu/.
[4] www.nltk.org/howto/wordnet.html.

Specifically, we extract the set of definitions $D_t = \{d_{tj}|j = 1, \ldots, m_t\}$ and the set of examples of use $E_t = \{e_{tj}|j = 1, \ldots, n_t\}$, each of them containing multiple words. Most of the senses are accompanied by a definition, but only about half of them also include an example of use.

Definitions d_{tj} and examples e_{tj} are represented by vectors defined as the average of the word embeddings over all the words constituting them (except stopwords). Formally, these vectors are $\mathbf{d}_{tj} = (\sum_{w_l \in d_{tj}} \mathbf{w}_l)/|d_{tj}|$ and $\mathbf{e}_{tj} = (\sum_{w_l \in e'_{tj}} \mathbf{w}_l)/|e'_{tj}|$, respectively, where $|d_{tj}|$ is the number of tokens of the definition. Although the entire definition d_{tj} is used to build the $\mathbf{d}_{tj}$ vector, we do not consider all words in the example e_{tj}, but limit the sum to a fragment e'_{tj} contained in a window of size c centered around the considered word, to avoid noise from long examples. Hence, we divide by the number of words in this window, noted $|e'_{tj}|$. All of these word vectors $\mathbf{w}_l$ are pre-trained word2vec embeddings from Google[5] (Mikolov et al., 2013). If dim is the dimensionality of the word vector space, then all vectors $\mathbf{w}_l$, $\mathbf{d}_{tj}$, and $\mathbf{e}_{tj}$ are in $\mathcal{R}^{dim}$. Each definition vector $\mathbf{d}_{tj}$ or example vector $\mathbf{e}_{tj}$ for a word type W_t is considered as a center vector for each sense during the clustering procedure.

Turning now to tokens, each word occurrence w_i in a source sentence is represented by the average vector $\mathbf{u}_i$ of the words from its context, that is, a window of c words centered on w_i, c being an even number. We calculate the vector $\mathbf{u}_i$ for w_i by averaging vectors from $c/2$ words before w_i and from $c/2$ words after it. We stop nevertheless at the sentence boundaries, and filter out stopwords before averaging.

2.2 Clustering Word Occurrences by Sense

We adapt three clustering algorithms to our needs for WSD applied to NMT. The objective is to cluster all occurrences w_i of a given word type W_t, represented as word vectors $\mathbf{u}_i$, according to the similarity of their senses, as inferred from the similarity of the context vectors. We compare the algorithms empirically in §5.

***K*-means Clustering.** The original k-means algorithm (MacQueen, 1967) aims to partition a set of items, which are here tokens $w_1, w_2, \ldots, w_n$ of the same word type W_t, represented through their embeddings $\mathbf{u}_1, \mathbf{u}_2, \ldots, \mathbf{u}_n$ where $\mathbf{u}_i \in \mathcal{R}^{dim}$. The goal of k-means is to partition (or cluster) these vectors into k sets $S = \{S_1, S_2, \ldots, S_k\}$ so as to minimize the within-cluster sum of squared distances to each centroid $\boldsymbol{\mu}_i$:

$$S = \arg\min_S \sum_{i=1}^{k} \sum_{\mathbf{u} \in S_i} ||\mathbf{u} - \boldsymbol{\mu}_i||^2 \qquad (1)$$

At the first iteration, when there are no clusters yet, the algorithm selects k random points as centroids of the k clusters. Then, at each subsequent iteration t, the algorithm calculates for each candidate cluster a new centroid of the observations, defined as their average vector, as follows:

$$\boldsymbol{\mu}_i^{t+1} = \frac{1}{|S_i^t|} \sum_{\mathbf{u}_j \in S_i^t} \mathbf{u}_j \qquad (2)$$

In an earlier application of k-means to phrase-based statistical MT, but not neural MT, we made several modifications to the original k-means algorithm to make it adaptive to the word senses observed in training data (Pu et al., 2017). We maintain these changes and summarize them briefly here. The initial number of clusters k_t for each ambiguous word type W_t is set to the number of its senses in WordNet, either considering only the senses that have a definition or those that have an example. The centroids of the clusters are initialized to the vectors representing the senses from WordNet, either using their definition vectors $\mathbf{d}_{tj}$ or their example vectors $\mathbf{e}_{tj}$. These initializations are thus a form of weak supervision of the clustering process.

Finally, and most importantly, after running the k-means algorithm, the number of clusters for each word type is reduced by removing the clusters that contain fewer than 10 tokens and assigning their tokens to the closest large cluster. "Closest" is defined in terms of the cosine distance between $\mathbf{u}_i$ and their centroids. The final number of clusters thus depends on the observed occurrences in the training data (which are the same data as for MT), and avoids modeling infrequent senses that are difficult to translate anyway. When used in NMT, in order to assign each new token from the test data to a cluster (i.e., to perform WSD), we select the closest centroid, again in terms of cosine distance.

Chinese Restaurant Process. The Chinese Restaurant Process (CRP) is an unsupervised

[5] code.google.com/archive/p/word2vec/.

method considered as a practical interpretation of a Dirichlet process (Ferguson, 1973) for non-parametric clustering. In the original analogy, each token is compared to a customer in a restaurant, and each cluster is a table where customers can be seated. A new customer can choose to sit at a table with other customers, with a probability proportional to the numbers of customers at that table, or sit at a new, empty table. In an application to multisense word embeddings, Li and Jurafsky (2015) proposed that the probability to "sit at a table" should also depend on the contextual similarity between the token and the sense modeled by the table. We build upon this idea and bring several modifications that allow for an instantiation with sense-related knowledge from WordNet, as follows.

For each word type W_t appearing in the data, we start by fixing the maximal number k_t of senses or clusters as the number of senses of W_t in WordNet. This avoids an unbounded number of clusters (as in the original CRP algorithm) and the risk of cluster sparsity by setting a non-arbitrary limit based on linguistic knowledge. Moreover, we define the initial centroid of each cluster as the word vector corresponding either to the definition $\mathbf{d}_{tj}$ of the respective sense, or alternatively to the example $\mathbf{e}_{tj}$ illustrating the sense.

For each token w_i and its context vector $\mathbf{u}_i$ the algorithm decides whether the token is assigned to one of the sense clusters S_j to which previous tokens have been assigned, or whether it is assigned to a new empty cluster, by selecting the option that has the highest probability, which is computed as follows:

$$P \propto \begin{cases} N_j(\lambda_1 s(\mathbf{u}_i, \mathbf{d}_{tj}) + \lambda_2 s(\mathbf{u}_i, \boldsymbol{\mu}_j)) \\ \quad \text{if } N_j \neq 0 \text{ (non-empty sense)} \\ \gamma s(\mathbf{u}_i, \mathbf{d}_{tj}) \\ \quad \text{if } N_j = 0 \text{ (empty sense)} \end{cases} \quad (3)$$

In other words, for a non-empty sense, the probability is proportional to the popularity of the sense (number of tokens it already contains, N_j) and to the weighted sum of two cosine similarities $s(\cdot, \cdot)$: one between the context vector $\mathbf{u}_i$ of the token and the definition of the sense $\mathbf{d}_{tj}$, and another one between $\mathbf{u}_i$ and the average context vector of the tokens already assigned to the sense ($\boldsymbol{\mu}_j$). These terms are weighted by the two hyper-parameters λ_1 and λ_2. For an empty sense, only the second term is used, weighted by the γ hyper-parameter.

Random Walks. Finally, we also consider for comparison a WSD method based on random walks on the WordNet knowledge graph (Agirre and Soroa, 2009; Agirre et al., 2014) available from the UKB toolkit.[6] In the graph, senses correspond to nodes and the relationships or dependencies between pairs of senses correspond to the edges between those nodes. From each input sentence, we extract its content words (nouns, verbs, adjectives, and adverbs) that have an entry in the WordNet weighted graph. The method calculates the probability of a random walk over the graph from a target word's sense ending on any other sense in the graph, and determines the sense with the highest probability for each analyzed word. In this case, the random walk algorithm is PageRank (Grin and Page, 1998), which computes a relative structural importance or "rank" for each node.

3 Integration with Neural MT

3.1 Baseline Neural MT Model

We now present several models integrating WSD into NMT, starting from an attention-based NMT baseline (Bahdanau et al., 2015; Luong et al., 2015). Given a source sentence X with words w^x, $X = (w_1^x, w_2^x, ..., w_T^x)$, the model computes a conditional distribution over translations, expressed as $p(Y = (w_1^y, w_2^y, ..., w_{T'}^y)|X)$. The neural network model consists of an encoder, a decoder, and an attention mechanism. First, each source word $w_t^x \in V$ is projected from a one-hot word vector into a continuous vector space representation $\mathbf{x}_t$. Then, the resulting sequence of word vectors is read by the bidirectional encoder, which consists of forward and backward recurrent networks (RNNs). The forward RNN reads the sequence in left-to-right order (i.e., $\overrightarrow{\mathbf{h}}_t = \overrightarrow{\phi}(\overrightarrow{\mathbf{h}}_{t-1}, \mathbf{x_t})$), and the backward RNN reads it right-to-left ($\overleftarrow{\mathbf{h}}_t = \overleftarrow{\phi}(\overleftarrow{\mathbf{h}}_{t+1}, \mathbf{x_t})$).

The hidden states from the forward and backward RNNs are concatenated at each time step t to form an "annotation" vector $\mathbf{h_t} = [\overrightarrow{\mathbf{h_t}}; \overleftarrow{\mathbf{h_t}}]$. Taken over several time steps, these vectors form the "context"—that is, a tuple of annotation vectors $C = (\mathbf{h_1}, \mathbf{h_2}, ..., \mathbf{h_T})$. The recurrent activation

[6] ixa2.si.ehu.es/ukb. Strictly speaking, this is the only genuine WSD method, as the two previous ones pertain to sense induction rather than disambiguation. However, for simplicity, we will refer to all of them as WSD.

functions $\overrightarrow{\phi}$ and $\overleftarrow{\phi}$ are either long short-term memory units (LSTM) or gated recurrent units (GRU).

The decoder RNN maintains an internal hidden state $z_{t'}$. After each time step t', it first uses the attention mechanism to weight the annotation vectors in the context tuple C. The attention mechanism takes as input the previous hidden state of the decoder and one of the annotation vectors, and returns a relevance score $e_{t',t} = f_{\text{ATT}}(\mathbf{z_{t'-1}}, \mathbf{h_t})$. These scores are normalized to obtain attention scores:

$$\alpha_{t',t} = exp(e_{t',t}) / \sum_{k=1}^{T} exp(e_{t',k}) \quad (4)$$

These scores serve to compute a weighted sum of annotation vectors $\mathbf{c_{t'}} = \sum_{t=1}^{T} \alpha_{t',t}\mathbf{h}_t$, which are used by the decoder to update its hidden state:

$$z_{t'} = \phi_z(z_{t'-1}, \mathbf{y}_{t'-1}, \mathbf{c}_{t'}) \quad (5)$$

Similarly to the encoder, ϕ_z is implemented as either an LSTM or GRU and $\mathbf{y}_{t'-1}$ is the target-side word embedding vector corresponding to word w^y.

3.2 Sense-aware Neural MT Models

To model word senses for NMT, we concatenate the embedding of each token with a vector representation of its sense, either obtained from one of the clustering methods presented in §2, or learned during encoding, as we will explain. In other words, the new vector $\mathbf{w'}_i$ representing each source token w_i consists of two parts: $\mathbf{w'}_i = [\mathbf{w}_i \ ; \ \boldsymbol{\mu}_i]$, where $\mathbf{w}_i$ is the word embedding learned by the NMT, and $\boldsymbol{\mu}_i$ is the sense embedding obtained from WSD or learned by the NMT. To represent these senses, we create two dictionaries, one for words and the other one for sense labels, which will be embedded in a low-dimensional space, before the encoder. We propose several models for using and/or generating sense embeddings for NMT, named and defined as follows.

Top Sense (*TOP*). In this model, we directly use the sense selected for each token by one of the WSD systems above, and use the embeddings of the respective sense as generated by NMT after training.

Weighted Average of Senses (*AVG*). Instead of fully trusting the decision of a WSD system (even one adapted to MT), we consider all listed senses and the respective cluster centroids learned by the WSD system. Then we convert the distances d_l between the input token vector and the centroid of each sense S_l into a normalized weight distribution either by a linear or a logistic normalization:

$$\omega_j = \frac{1 - d_j}{\sum_{1 \le l \le k} d_l} \text{ or } \omega_j = \frac{e^{-d_j^2}}{\sum_{1 \le l \le k} e^{-d_l^2}} \quad (6)$$

where k is the total number of senses of token w_i. The sense embedding for each token is computed as the weighted average of all sense embeddings:

$$\boldsymbol{\mu}_i = \sum_{1 \le j \le k} \omega_j \boldsymbol{\mu}_{ij} \quad (7)$$

Attention-Based Sense Weights (*ATT*). Instead of obtaining the weight distribution from the centroids computed by WSD, we also propose to dynamically compute the probability of relatedness to each sense based on the current word and sense embeddings during encoding, as follows. Given a token w_i, we consider all the other tokens in the sentence $(w_1, \ldots, w_{i-1}, w_{i+1}, \ldots, w_L)$ as the context of w_i, where L is the length of the sentence. We define the context vector of w_i as the mean of all the embeddings $\boldsymbol{u}_j$ of the words w_j, that is, $\mathbf{u}_i = (\sum_{l \neq i} \boldsymbol{u}_l)/(L-1)$. Then, we compute the similarity $f(\mathbf{u}_i, \boldsymbol{\mu}_{ij})$ between each sense embedding $\boldsymbol{\mu}_{ij}$ and the context vector $\mathbf{u}_i$ using an additional attention layer in the network, with two possibilities that will be compared empirically:

$$f(\mathbf{u}_i, \boldsymbol{\mu}_{ij}) = v^T tanh(W\mathbf{u}_i + U\boldsymbol{\mu}_{ij}) \quad (8)$$

or

$$f(\mathbf{u}_i, \boldsymbol{\mu}_{ij}) = \mathbf{u}_i^T W \boldsymbol{\mu}_{ij} \quad (9)$$

The weights ω_j are now obtained through the following softmax normalization:

$$\omega_j = \frac{e^{f(\mathbf{u}_i, \boldsymbol{\mu}_{ij})}}{\sum_{1 \le l \le k} e^{f(\mathbf{u}_i, \boldsymbol{\mu}_{il})}} \quad (10)$$

Finally, the average sense embedding is obtained as in Equation (7), and is concatenated to the word vector $\mathbf{u}_i$.

***ATT* Model with Initialization of Embeddings (*ATT$_{ini}$*).** The fourth model is similar to the *ATT* model, with the difference that we initialize the embeddings of the source word dictionary using the word2vec vectors of the word types, and the embeddings of the sense dictionary using the centroid vectors obtained from k-means.

TL	Train	Dev	Test	Labels		Words
				Nouns	Verbs	
FR	0.5M	5k	50k	3,910	1,627	2,006
	5.3M	4,576	6,003	8,276	3,059	3,876
DE	0.5M	5k	50k	3,885	1,576	1,976
	4.5M	3,000	5,172	7,520	1,634	3,194
ES	0.5M	5k	50k	3,862	1,627	1,987
	3.9M	4,576	6,003	7,549	2,798	3,558
ZH	0.5M	5K	50K	3,844	1,475	1,915
NL	0.5M	5K	50K	3,915	1,647	2,210

Table 1: Size of data sets used for machine translation from English to five different target languages (TL). FR = French; DE = German; ES = Spanish; ZH = Chinese; NL = Dutch.

4 Data, Metrics, and Implementation

Data Sets. We train and test our sense-aware MT systems on the data shown in Table 1: the UN Corpus[7] (Rafalovitch and Dale, 2009) and the Europarl Corpus[8] (Koehn, 2005). We first experiment with our models using the same data set and protocol as in our previous work (Pu et al., 2017), to enable comparisons with phrase-based statistical MT systems, for which the sense of each ambiguous source word was modeled as a factor. Moreover, in order to make a better comparison with other related approaches, we train and test our sense-aware NMT models on large data sets from Workshop on Statistical Machine Translation (WMT) shared tasks over three language pairs (EN/DE, EN/ES, and EN/FR).

The data set used in our previous work consists of 500k parallel sentences for each language pair, 5k for development and 50k for testing. The data originates from UN for EN/ZH, and from Europarl for the other pairs. The source sides of these sets contain around 2,000 different English word forms (after lemmatization) that have more than one sense in WordNet. Our WSD system generates ca. 3.8K different noun labels and 1.5K verb labels for these word forms.

The WMT data sets additionally used in this paper are the following ones. First, we use the complete EN/DE set from WMT 2016 (Bojar et al., 2016) with a total of ca. 4.5M sentence pairs. In this case, the development set is NewsTest 2013, and the testing set is made of NewsTest 2014 and 2015. Second, for EN/FR and EN/ES, we use data from WMT 2014 (Bojar et al., 2014)[9] with 5.3M sentences for EN/FR and 3.8M sentences for EN/ES. Here, the development sets are NewsTest 2008 and 2009, and the testing sets are NewsTest 2012 and 2013 for both language pairs. The source sides of these larger additional sets contain around 3,500 unique English word forms with more than one sense in WordNet, and our system generates ca. 8K different noun labels and 2.5K verb labels for each set.

Finally, for comparison purposes and model selection, we use the WIT^3 Corpus[10] (Cettolo et al., 2012), a collection of transcripts of TED talks. We use 150K sentence pairs for training, 5K for development and 50K for testing.

Pre-processing. Before assigning sense labels, we tokenize all the texts and identify the parts of speech using the Stanford POS tagger.[11] Then, we filter out the stopwords and the nouns that are proper names according to the Stanford Name Entity Recognizer.[11] Furthermore, we convert the plural forms of nouns to their singular forms and the verb forms to infinitives using the stemmer and lemmatizer from NLTK,[12] which is essential because WordNet has description entries only for base forms. The pre-processed text is used for assigning sense labels to each occurrence of a noun or verb that has more than one sense in WordNet.

***K*-means Settings.** Unless otherwise stated, we adopt the following settings in the k-means algorithm, with the implementation provided in Scikit-learn (Pedregosa et al., 2011). We use the definition of each sense for initializing the centroids, and later compare this choice with the use of examples. We set k_t, the initial number of clusters, to the number of WordNet senses of each ambiguous word type W_t, and set the window size for the context surrounding each occurrence to $c = 8$.

Neural MT. We build upon the attention-based neural translation model (Bahdanau et al., 2015) from the OpenNMT toolkit (Klein et al., 2017).[13] We use LSTM and not GRU. For the proposed *ATT* and ATT_{ini} models, we add an

[7] www.uncorpora.org.
[8] www.statmt.org/europarl.
[9] We selected the data from different years of WMT because the EN/FR and EN/ES pairs were only available in WMT 2014.
[10] wit3.fbk.eu.
[11] nlp.stanford.edu/software.
[12] www.nltk.org.
[13] www.opennmt.net.

external attention layer before the encoder, but do not otherwise alter the internals of the NMT model.

We set the source and target vocabulary sizes to 50,000 and the dimension of word embeddings to 500, which is recommended for OpenNMT, so as to reach a strong baseline. For the ATT_{ini} model, because the embeddings from word2vec used for initialization have only 300 dimensions, we randomly pick up a vector with 200 dimensions within range [−0.1,0.1] and concatenate it with the vector from word2vec to reach the required number of dimensions, ensuring a fair comparison.

It takes around 15 epochs (25–30 hours on Idiap's GPU cluster) to train each of the five NMT models: the baseline and our four proposals. The *AVG* model takes more time for training (around 40 hours) because we use additional weights and senses for each token. In fact, we limit the number of senses for *AVG* to 5 per word type, after observing that in WordNet there are fewer than 100 words with more than 5 senses.

Evaluation Metrics. For the evaluation of intrinsic WSD performance, we use the V-score, the F_1-score, and their average, as used for instance at SemEval 2010 (Manandhar et al., 2010). The V-score is the weighted harmonic mean of homogeneity and completeness (favoring systems generating more clusters than the reference), and the F_1-score measures the classification performance (favoring systems generating fewer clusters). Therefore, the ranking metric for SemEval 2010 is the average of the two.

We select the optimal model configuration based on MT performance on development sets, as measured with the traditional *multi-bleu* score (Papineni et al., 2002). Moreover, to estimate the impact of WSD on MT, we also measure the actual impact on the nouns and verbs that have several WordNet senses, by counting how many of them are translated exactly as in the reference translation. To quantify the difference with the baseline, we use the following coefficient. First, for a certain set of tokens in the source data, we note as N_{improved} the number of tokens that are translated by our system with the same token as in the reference translation, but are translated differently by the baseline system. Conversely, we note as N_{degraded} the number of tokens that are translated by the baseline system as in the reference, but differently by our system.[14] We use the normalized coefficient $\rho = (N_{\text{improved}} - N_{\text{degraded}})/T$, where T is the total number of tokens, as a metric to specifically evaluate the translation of words submitted to WSD.

For all tables we mark in bold the best score per condition. For MT scores in Tables 5, 7, and 8, we show the improvement over the baseline and its significance based on two confidence levels: either $p < 0.05$ (indicated with a '†') or $p < 0.01$ ('‡'). Any p-values larger than 0.05 are treated as not significant and are left unmarked.

5 Optimal Values of the Parameters

5.1 Best WSD Method Based on BLEU

We first select the optimal clustering method and its initialization settings, in a series of experiments with statistical MT over the WIT^3 corpus, extending and confirming our previous results (Pu et al., 2017). In Table 2, we present the BLEU and ρ scores of our previous WSD+SMT system for the three clustering methods, initialized with vectors either from the WordNet definitions or from examples, for two language pairs. We also provide BLEU scores of baseline systems and of oracle ones (i.e., using correct senses as factors). The best method is k-means and the best initialization is with the vectors of definitions. All values of ρ show improvements over the baseline, with up to 4% for k-means on DE/EN.

Moreover, we found that random initializations underperform with respect to definitions or examples. For a fair comparison, we set the number of clusters equal either to the number of synsets with definitions or with examples, for each word type, and obtained BLEU scores on EN/ZH of 15.34 and 15.27, respectively—hence lower than 15.54 and 15.41 in Table 2. We investigated earlier (Pu et al., 2017) the effect of the context window surrounding each ambiguous token, and found with the WSD+SMT factored system on EN/ZH WIT^3 data that the optimal size was 8, which we use here as well.

5.2 Best WSD Method Based on V/F1 Scores

Table 3 shows our WSD results in terms of V-score and F_1-score, comparing our methods (six

[14] The values of N_{improved} and N_{degraded} are obtained using automatic word alignment. They do not capture, of course, the intrinsic correctness of a candidate translation, but only its identity or not with one reference translation.

Pair	Initialization	BLEU					ρ (%)		
		Baseline	Graph	CRP	k-means	Oracle	Graph	CRP	k-means
EN/ZH	Definitions	15.23	15.31	15.31	**15.54**	16.24	+0.20	+0.27	**+2.25**
	Examples			15.28	15.41	15.85		+0.13	+1.60
EN/DE	Definitions	19.72	19.74	19.69	**20.23**	20.99	−0.07	−0.19	**+3.96**
	Examples			19.74	19.87	20.45		−0.12	+2.15

Table 2: Performance of the WSD+SMT factored system for two language pairs from WIT3, with three clustering methods and two initializations.

System	V-score			F_1-score			Average			C
	All	Nouns	Verbs	All	Nouns	Verbs	All	Nouns	Verbs	
UoY	15.70	20.60	8.50	49.80	38.20	66.60	32.75	29.40	**37.50**	11.54
KCDC-GD	6.90	5.90	8.50	59.20	51.60	70.00	33.05	28.70	39.20	2.78
Duluth-Mix-Gap	3.00	2.90	3.00	59.10	54.50	65.80	31.05	29.70	34.40	1.61
k-means+definitions	13.65	14.70	12.60	56.70	53.70	59.60	**35.20**	**34.20**	36.10	4.45
k-means+examples	11.35	11.00	11.70	53.25	47.70	58.80	32.28	29.30	35.25	3.58
CRP + definitions	1.45	1.50	1.45	64.80	56.80	72.80	33.13	29.15	37.10	1.80
CRP + examples	1.20	1.30	1.10	64.75	56.80	72.70	32.98	29.05	36.90	1.66
Graph + definitions	11.30	11.90	10.70	55.10	52.80	57.40	33.20	32.35	34.05	2.63
Graph + examples	9.05	8.70	9.40	50.15	45.20	55.10	29.60	26.96	32.25	2.08

Table 3: WSD results from three SemEval 2010 systems and our six systems, in terms of V-score, F_1 score, and their average. C = the average number of clusters. The adaptive k-means using definitions outperforms the others on the average of V and F_1, when considering both nouns and verbs, or nouns only. The SemEval systems are UoY (Korkontzelos and Manandhar, 2010); KCDC-GD (Kern et al., 2010); and Duluth-Mix-Gap (Pedersen, 2010).

lines at the bottom) with other significant systems that participated in the SemEval 2010 shared task (Manandhar et al., 2010).[15] The adaptive k-means initialized with definitions has the highest average score (35.20) and ranks among the top systems for most of the metrics individually. Moreover, the adaptive k-means method finds on average 4.5 senses per word type, which is very close to the ground-truth value of 4.46. Overall, we observed that k-means infers fewer senses per word type than WordNet. These results show that k-means WSD is effective and provides competitive performance against other weakly supervised alternatives (CRP or Random Walk) and even against SemEval WSD methods, but using additional knowledge not available to SemEval participants.

5.3 Selection of WSD+NMT Model

To compare several options of the WSD+NMT systems, we trained and tested them on a subset of EN/FR Europarl (a smaller data set shortened the training times). The results are shown in Table 4. For the *AVG* model, the logistic normalization in Equation (6) works better than the linear one. For the *ATT* model, we compared two different labeling approaches for tokens that do not have multiple senses: Either use the same NULL label for all tokens, or use the word itself as a label for its sense; the second option appeared to be the best. Finally, for the ATT_{ini} model, we compared the two options for the attention function in Equation (8), and found that the formula with $tanh$ is the best. In what follows, we use these settings for the *AVG* and *ATT* systems.

System and settings	BLEU
Baseline	29.55
TOP	29.63 *(+0.08)*
AVG with linear norm. in Eq. (6)	29.67 *(+0.12)*
AVG with logistic norm. in Eq. (6)	**30.15** *(+0.60)*
ATT with NULL label	29.80 *(+0.33)*
ATT with word used as label	**30.23** *(+0.68)*
ATT_{ini} with $\mathbf{u}_i^T W \boldsymbol{\mu}_{ij}$ in Eq. (8)	29.94 *(+0.39)*
ATT_{ini} with $tanh$ in Eq. (8)	**30.61** *(+1.06)*

Table 4: Performance of various WSD+NMT configurations on a EN/FR subset of Europarl, with variations with respect to baseline. We select the settings with the best performance (**bold**) for our final experiments in §6.

[15] We provide comparisons with more systems from SemEval in our previous paper (Pu et al., 2017).

	EN/FR	EN/DE	EN/ZH	EN/ES	EN/NL
SMT baseline	31.96	20.78	23.25	39.95	23.56
Graph	32.01 *(+.05)*	21.17 *(+.39)*	23.47 *(+.22)*	40.15 *(+.20)*	23.74 *(+.18)*
CRP	32.08 *(+.12)*	21.19 *(+.41)* †	23.55 *(+.29)*	40.14 *(+.19)*	23.79 *(+.23)*
k-means	32.20 *(+.24)*	21.32 *(+.54)* †	23.69 *(+.44)* †	40.37 *(+.42)* †	23.84 *(+.26)*
NMT baseline	34.60	25.80	27.07	44.09	24.79
k-means + *TOP*	34.52 *(−.08)*	25.84 *(+.04)*	26.93 *(−.14)*	44.14 *(+.05)*	24.71 *(−.08)*
k-means + *AVG*	35.17 *(+.57)* †	26.47 *(+.67)* †	27.44 *(+.37)*	45.05 *(+.97)* ‡	25.04 *(+.25)*
None + *ATT*	35.32 *(+.72)* ‡	26.50 *(+.70)* ‡	27.56 *(+.49)* †	44.93 *(+.84)* ‡	25.36 *(+.57)* †
k-means + ATT_{ini}	**35.78** ***(+1.18)*** ‡	**26.74** ***(+.94)*** ‡	**27.84** ***(+.77)*** ‡	**45.18** ***(+1.09)*** ‡	**25.65** ***(+.86)*** ‡

Table 5: BLEU scores of our sense-aware NMT systems over five language pairs: ATT_{ini} is the best one among SMT and NMT systems. Significance testing is indicated by † for $p < 0.05$ and ‡ for $p < 0.01$.

6 Results

We first evaluate our sense-aware models with smaller data sets (ca. 500K lines) for five language pairs with English as source. We evaluate them through both automatic measures and human assessment. Later on, we evaluate our sense-aware NMT models with larger WMT data sets to enable a better comparison with other related approaches.

BLEU scores. Table 5 displays the performance of both sense-aware phrase-based and neural MT systems with the training sets of 500K lines listed in Table 1 on five language pairs. Specifically, we compare several approaches that integrate word sense information in SMT and NMT. The best hyper-parameters are those found above, for each of the WSD+NMT combination strategies, in particular the k-means method for WSD+SMT, and the ATT_{ini} method for WSD+NMT—that is, the attention-based model of senses initialized with the output of k-means clustering.

Comparisons with Baselines. Table 5 shows that our WSD+NMT systems perform consistently better than the baselines, with the largest improvements achieved by NMT on EN/FR and EN/ES. The neural systems outperform the phrase-based statistical ones (Pu et al., 2017), which are shown for comparison in the upper part of the table.

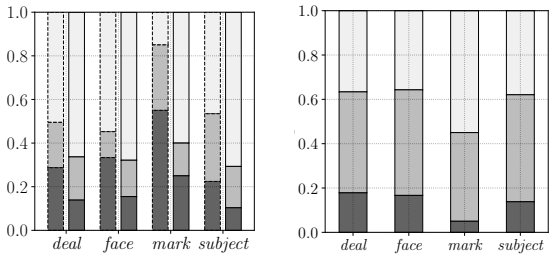
We compare our proposal to the recent system proposed by Yang et al. (2017), on the 500K-line EN/FR Europarl data set (the differences between their system and ours are listed in §7). We carefully implemented their model by following their paper, since their code is not available. Using the sense embeddings of the multi-sense skip-gram model (MSSG) (Neelakantan et al., 2014) as they do, and training for six epochs as in their study, our implementation of their model reaches only 31.05 BLEU points. When increasing the training stage until convergence (15 epochs), the best BLEU score is 34.52, which is still below our NMT baseline of 34.60. We also found that the initialization of embeddings with MSSG brings less than 1 BLEU point improvement with respect to random initializations (which scored 30.11 over six epochs and 33.77 until convergence), while Yang et al. found a 1.3–2.7 increase on two different test sets. In order to better understand the difference, we tried several combinations of their model with ours. We obtain a BLEU score of 35.02 by replacing their MSSG sense specification model with our adaptive k-means approach, and a BLEU score of 35.18 by replacing our context calculation method (averaging word embeddings within one sentence) with their context vector generation method, which is computed from the output of a bi-directional RNN. In the end, the best BLEU score on this EN/FR data set (35.78 as shown in Table 5, column 1, last line) is reached by our system with its best options.

Lexical Choice. Using word alignment, we assess the improvement brought by our systems with respect to the baseline in terms of the number of words—here, WSD-labeled nouns and verbs—that are translated exactly as in the reference translation (modulo alignment errors). These numbers can be arranged in a confusion matrix with four values: the words translated correctly (i.e., as in the reference) by both systems, those translated correctly by one system but incorrectly by the other one, and vice versa, and those translated incorrectly by both.

Table 6 shows the confusion matrix for our sense-aware NMT with the ATT_{ini} model versus

		Baselines			
		EN/FR		EN/ES	
		Correct	Incorrect	Correct	Incorrect
WSD+ NMT	C.	134,552	17,145	146,806	16,523
	I.	10,551	101,228	8,183	58,387
WSD+ SMT	C.	124,759	13,408	139,800	11,194
	I.	9,676	115,633	7,559	71,346

Table 6: Confusion matrix for our WSD+NMT (ATT_{ini}) system and our WSD+SMT system against their respective baselines (NMT and SMT), over the Europarl test data, for two language pairs.



(a) System ratings. (b) Comparative scores.

Figure 1: Human comparison of the EN/FR translations of four word types. (a) Proportion of good (light gray), acceptable (middle gray), and wrong (dark gray) translations per word and system (baseline left, ATT_{ini} right, for each word). (b) Proportion of translations in which ATT_{ini} is better (light gray), equal (middle gray), or worse (dark gray) than the baseline.

the NMT baseline over the Europarl test data. The net improvement (i.e., the fraction of words improved by our system minus those degraded[16]) appears to be +2.5% for EN/FR and +3.6% for EN/ES. For comparison, we show the results of the WSD+SMT system versus the SMT baseline in the lower part of Table 6: The improvement is smaller, at +1.4% for EN/FR and +1.5% for EN/ES. Therefore, the ATT_{ini} NMT model brings higher benefits over the NMT baseline than the WSD+SMT factored model, although the NMT baseline is stronger than the SMT one (see Table 5).

Human Assessment. To compare our systems against baselines, we also consider a human evaluation of the translation of words with multiple senses (nouns or verbs). The goal is to capture more precisely the correct translations that are, however, different from the reference.

Given the cost of the procedure, one evaluator with good knowledge of EN and FR rated the translations of four word types that appear frequently in the test set and have multiple possible senses and translations into French. These words are: deal (101 tokens), face (84), mark (20), and subject (58). Two translations of deal are exemplified in §1.

For each occurrence, the evaluator sees the source sentence, the reference translation, and the outputs of the NMT baseline and the ATT_{ini} in random order, so that the system cannot be identified. The two translations of the considered word are rated as good, acceptable, or wrong. We submit only cases in which the two translations differ, to minimize the annotation effort with no impact on the comparison between systems.

[16]Explicitly, improvements are (system-correct & baseline-incorrect) minus (system-incorrect & baseline-correct), and degradations the converse difference.

Firstly, Figure 1(a) shows that ATT_{ini} has a higher proportion of good translations, and a lower proportion of wrong ones, for all four words. The largest difference is for subject, where ATT_{ini} has 75% good translations and the baseline only 46%; moreover, the baseline has 22% errors and ATT_{ini} has only 9%. Secondly, Figure 1(b) shows the proportions of tokens, for each type, for which ATT_{ini} was respectively better, equal, or worse than the baseline. Again, for each of the four words, there are far more improvements brought by ATT_{ini} than degradations. On average, 40% of the occurrences are improved and only 10% are degraded.

Results on WMT Data Sets. To demonstrate that our findings generalize to larger data sets, we report results on three data sets provided by the WMT conference (see §4), namely, for EN/DE, EN/ES and EN/FR. Tables 7 and 8 show the results of our proposed NMT models on these test sets. The results in Table 7 confirm that our sense-aware NMT models improve significantly the translation quality also on larger data sets, which permit stronger baselines. Comparing these results with the ones from Table 5, we even conclude that our models trained on larger, mixed-domain data sets achieve higher improvements than the models trained on smaller, domain-specific data sets (Europarl). This clearly shows that our sense-aware NMT models are beneficial on both narrow and broad domains.

Finally, we compare our model with several recent NMT models that make use of contextual information, thus sharing a similar overall goal to our study. Indeed, the model proposed by

	EN/FR		EN/ES	
	NT12	NT13	NT12	NT13
Baseline	29.09	29.60	32.66	29.57
None + *ATT*	29.47 *(+.38)*	30.21 *(+.61)* †	33.15 *(+.49)* †	30.27 *(+.70)* ‡
k-means + ATT_{ini}	**30.26** *(+1.17)* ‡	**30.95** *(+.1.35)* ‡	**34.14** *(+1.48)* ‡	**30.67** *(+1.1)* ‡

Table 7: BLEU scores on WMT NewsTest 2012 and 2013 (NT) test sets for two language pairs. Significance testing is indicated by † for $p < 0.05$ and ‡ for $p < 0.01$.

NMT model	NT14	NT15
Context-dependent (Choi et al., 2017)	-	21.99
Context-aware (Zhang et al., 2017)	22.57	-
Self-attentive (Werlen et al., 2018)	23.2	25.5
Baseline	22.79	24.94
None + *ATT*	23.34 †	25.28
k-means + ATT_{ini}	**23.85** *(+1.14)* ‡	**25.71** *(+0.77)* ‡

Table 8: BLEU score on English-to-German translation over the WMT NewsTest (NT) 2014 and 2015 test sets. Significance testing is indicated by † for $p < 0.05$ and ‡ for $p < 0.01$. The highest score per column is in **bold**.

Choi et al. (2017) attempts to improve NMT by integrating context vectors associated to source words into the generation process during decoding. The model proposed by Zhang et al. (2017) is aware of previous attended words on the source side in order to better predict which words will be attended in future. The self-attentive residual decoder designed by Werlen et al. (2018) leverages the contextual information from previously translated words on the target side. BLEU scores on the English–German pair shown in Table 8 demonstrate that our baseline is strong and that our model is competitive with respect to recent models that leverage contextual information in different ways.

7 Related Work

Word sense disambiguation aims to identify the sense of a word appearing in a given context (Agirre and Edmonds, 2007). Resolving word sense ambiguities should be useful, in particular, for lexical choice in MT. An initial investigation found that a statistical MT system that makes use of off-the-shelf WSD does not yield significantly better quality translations than an SMT system not using it (Carpuat and Wu, 2005). However, several studies (Cabezas and Resnik, 2005; Vickrey et al., 2005; Carpuat and Wu, 2007; Chan et al., 2007) reformulated the task of WSD for SMT and showed that integrating the ambiguity information generated from modified WSD improved SMT by 0.15–0.57 BLEU points compared with baselines.

Recently, Tang et al. (2016) used only the supersenses from WordNet (coarse-grained semantic labels) for automatic WSD, using maximum entropy classification or sense embeddings learned using word2vec. When combining WSD with SMT using a factored model, Tang et al. improved BLEU scores by 0.7 points on average, though with large differences between their three test subsets (IT Q&A pairs).

Although these reformulations of the WSD task proved helpful for SMT, they did not determine whether actual source-side senses are helpful or not for end-to-end SMT. Xiong and Zhang (2014) attempted to answer this question by performing self-learned word sense induction instead of using pre-specified word senses as traditional WSD does. However, they created the risk of discovering sense clusters that do not correspond to the senses of words actually needed for MT. Hence, they left open an important question, namely, whether WSD based on semantic resources such as WordNet (Fellbaum, 1998) can be successfully integrated with SMT.

Several studies integrated sense information as features to SMT, either obtained from the sense graph provided by WordNet (Neale et al., 2016) or generated from both sides of word dependencies (Su et al., 2015). However, apart from the sense graph, WordNet also provides textual information such as sense definitions and examples, which should be useful for WSD, but were not

used in these studies. In previous work (Pu et al., 2017), we used this information to perform sense induction on source-side data using k-means and demonstrated improvement with factored phrase-based SMT but not NMT.

Neural MT became the state of the art (Sutskever et al., 2014; Bahdanau et al., 2015). Instead of working directly at the discrete symbol level as SMT, it projects and manipulates the source sequence of discrete symbols in a continuous vector space. However, NMT generates only one embedding for each word type, regardless of its possibly different senses, as analyzed, for example, by Hill et al. (2017). Several studies proposed efficient nonparametric models for monolingual word sense representation (Neelakantan et al., 2014; Li and Jurafsky, 2015; Bartunov et al., 2016; Liu et al., 2017), but left open the question whether sense representations can help neural MT by reducing word ambiguity. Recent studies integrate the additional sense assignment with neural MT based on these approaches, either by adding such sense assignments as additional features (Rios et al., 2017) or by merging the context information on both sides of parallel data for encoding and decoding (Choi et al., 2017).

Yang et al. (2017) recently proposed to add sense information by using weighted sense embeddings as input to neural MT. The sense labels were generated by a MSSG model (Neelakantan et al., 2014), and the context vector used for sense weight generation was computed from the output of a bidirectional RNN. Finally, the weighted average sense embeddings were used in place of the word embedding for the NMT encoder. The numerical results given in §6 show that our options for using sense embeddings outperform Yang et al.'s proposal. In fact, their approach even performed worse than the NMT baseline on our EN/FR data set. We conclude that adaptive k-means clustering is better than MSSG for use in NMT, and that concatenating the word embedding and its sense vector as input for the RNN encoder is better than just using the sense embedding for each token. In terms of efficiency, Yang et al. (2017) need an additional bidirectional RNN to generate the context vector for each input token, whereas we compute the context vector by averaging the embeddings of the neighboring tokens. This slows down the training of the encoder by a factor of 3, which may explain why they only trained their model for six epochs.

8 Conclusion

We presented a neural MT system enhanced with an attention-based method to represent multiple word senses, making use of a larger context to disambiguate words that have various possible translations. We proposed several adaptive context-dependent clustering algorithms for WSD and combined them in several ways with NMT—following our earlier experiments with SMT (Pu et al., 2017)—and found that they had competitive WSD performance on data from the SemEval 2010 shared task.

For NMT, the best-performing method used the output of k-means to initialize the sense embeddings that are learned by our system. In particular, it appeared that learning sense embeddings for NMT is better than using embeddings learned separately by other methods, although such embeddings may be useful for initialization. Our experiments with five language pairs showed that our sense-aware NMT systems consistently improve over strong NMT baselines, and that they specifically improve the translation of words with multiple senses.

In the future, our approach to sense-aware NMT could be extended to other NMT architectures such as the Transformer network proposed by Vaswani et al. (2017). As was the case with the LSTM-based architecture studied here, the Transformer network does not explicitly model or utilize the sense information of words, and, therefore, we hypothesize that its performance could also be improved by using our sense integration approaches. To encourage further research in sense-aware NMT, our code is made available at `https://github.com/idiap/sense_aware_NMT`.

Acknowledgments

The authors are grateful for support from the Swiss National Science Foundation through the MODERN Sinergia project on Modeling Discourse Entities and Relations for Coherent Machine Translation, grant no. 147653 (`www.idiap.ch/project/modern`), and from the European Union through the SUMMA Horizon 2020 project on Scalable Understanding of Multilingual Media, grant no. 688139 (`www.summa-project.eu`). The authors would like to thank the *TACL* editors and reviewers for their helpful comments and suggestions.

References

Eneko Agirre and Philip Edmonds. 2007. *Word Sense Disambiguation: Algorithms and Applications*. Springer-Verlag, Berlin, Germany.

Eneko Agirre, Oier López de Lacalle, and Aitor Soroa. 2014. Random walks for knowledge-based word sense disambiguation. *Computational Linguistics*, 40(1):57–84.

Eneko Agirre and Aitor Soroa. 2009. Personalizing PageRank for word sense disambiguation. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, pages 33–41, Athens, Greece.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *Proceedings of the International Conference on Learning Representations*, San Diego, CA, USA.

Sergey Bartunov, Dmitry Kondrashkin, Anton Osokin, and Dmitry Vetrov. 2016. Breaking sticks and ambiguities with adaptive skip-gram. In *Artificial Intelligence and Statistics*, pages 130–138.

Ondřej Bojar, Christian Buck, Christian Federmann, Barry Haddow, Philipp Koehn, Johannes Leveling, Christof Monz, Pavel Pecina, Matt Post, Herve Saint-Amand, Radu Soricut, Lucia Specia, and Aleš Tamchyna. 2014. Findings of the 2014 workshop on statistical machine translation. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 12–58, Baltimore, Maryland, USA.

Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, Varvara Logacheva, Christof Monz, Matteo Negri, Aurelie Neveol, Mariana Neves, Martin Popel, Matt Post, Raphael Rubino, Carolina Scarton, Lucia Specia, Marco Turchi, Karin Verspoor, and Marcos Zampieri. 2016. Findings of the 2016 conference on machine translation. In *Proceedings of the First Conference on Machine Translation*, pages 131–198, Berlin, Germany.

Clara Cabezas and Philip Resnik. 2005. Using WSD techniques for lexical selection in statistical machine translation. Technical report, DTIC Document.

Marine Carpuat and Dekai Wu. 2005. Word sense disambiguation vs. statistical machine translation. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*, pages 387–394, Michigan, MI, USA.

Marine Carpuat and Dekai Wu. 2007. Improving statistical machine translation using word sense disambiguation. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 61–72, Prague, Czech Republic.

Mauro Cettolo, Christian Girardi, and Marcello Federico. 2012. WIT^3: Web inventory of transcribed and translated talks. In *Proceedings of the 16th Conference of the European Association for Machine Translation (EAMT)*, pages 261–268, Trento, Italy.

Yee Seng Chan, Hwee Tou Ng, and David Chiang. 2007. Word sense disambiguation improves statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*, pages 33–40, Prague, Czech Republic.

Heeyoul Choi, Kyunghyun Cho, and Yoshua Bengio. 2017. Context-dependent word representation for neural machine translation. *Computer Speech & Language*, 45:149–160.

Christiane Fellbaum, editor. 1998. *WordNet: An Electronic Lexical Database*. The MIT Press, Cambridge, MA, USA.

Thomas S. Ferguson. 1973. A Bayesian analysis of some nonparametric problems. *The Annals of Statistics*, 1(2):209–230.

Sergey Grin and Lawrence Page. 1998. The anatomy of a large-scale hypertextual Web search engine. *Computer Networks and ISDN Systems*, 30(1-7):107–117.

Felix Hill, Kyunghyun Cho, Sébastien Jean, and Yoshua Bengio. 2017. The representational geometry of word meanings acquired by neural machine translation models. *Machine Translation*, 31(1):3–18.

Roman Kern, Markus Muhr, and Michael Granitzer. 2010. KCDC: Word sense induction by using grammatical dependencies and sentence phrase structure. In *Proceedings of the 5th International Workshop on Semantic Evaluation (SemEval-2010)*, pages 351–354, Los Angeles, CA, USA.

Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander M. Rush. 2017. OpenNMT: Open-source toolkit for neural machine translation. *CoRR*, abs/1701.02810v2.

Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *Proceedings of MT Summit X*, pages 79–86, Phuket, Thailand.

Ioannis Korkontzelos and Suresh Manandhar. 2010. UoY: graphs of ambiguous vertices for word sense induction and disambiguation. In *Proceedings of the 5th International Workshop on Semantic Evaluation (SemEval 2010)*, pages 355–358, Los Angeles, California.

Jiwei Li and Dan Jurafsky. 2015. Do multi-sense embeddings improve natural language understanding? In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1722–1732, Lisbon, Portugal.

Frederick Liu, Han Lu, and Graham Neubig. 2017. Handling homographs in neural machine translation. *CoRR*, abs/1708.06510v2.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1412–1421, Lisbon, Portugal.

James MacQueen. 1967. Some methods for classification and analysis of multivariate observations. In *Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability*, pages 281–297. Oakland, CA, USA.

Suresh Manandhar, Ioannis P. Klapaftis, Dmitriy Dligach, and Sameer S. Pradhan. 2010. SemEval-2010 task 14: Word sense induction and disambiguation. In *Proceedings of the 5th International Workshop on Semantic Evaluation (SemEval 2010)*, pages 63–68, Los Angeles, CA, USA.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems (NIPS)*, pages 3111–3119.

Andrea Moro and Roberto Navigli. 2015. Semeval-2015 task 13: Multilingual all-words sense disambiguation and entity linking. In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 288–297, Denver, Colorado. Association for Computational Linguistics.

Steven Neale, Luıs Gomes, Eneko Agirre, Oier Lopez de Lacalle, and António Branco. 2016. Word sense-aware machine translation: Including senses as contextual features for improved translation models. In *Proceedings of the 10th International Conference on Language Resources and Evaluation (LREC 2016)*, pages 2777–2783, Portoroz, Slovenia.

Arvind Neelakantan, Jeevan Shankar, Alexandre Passos, and Andrew McCallum. 2014. Efficient non-parametric estimation of multiple embeddings per word in vector space. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1059–1069, Doha, Qatar.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of Association for Computational Linguistics*, pages 311–318, Philadelphia, USA.

Ted Pedersen. 2010. Duluth-WSI: SenseClusters applied to the sense induction task of SemEval-2. In *Proceedings of the 5th International Workshop on Semantic Evaluation (SemEval 2010)*, pages 363–366, Los Angeles, CA, USA.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David

Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. 2011. Scikit-learn: Machine learning in Pmaython. *Journal of Machine Learning Research*, 12:2825–2830.

Xiao Pu, Nikolaos Pappas, and Andrei Popescu-Belis. 2017. Sense-aware statistical machine translation using adaptive context-dependent clustering. In *Proceedings of the Second Conference on Machine Translation*, pages 1–10, Copenhagen, Denmark.

Alexandre Rafalovitch and Robert Dale. 2009. United Nations general assembly resolutions: A six-language parallel corpus. In *Proceedings of MT Summit XII*, pages 292–299, Ontario, ON. Canada.

Annette Rios, Laura Mascarell, and Rico Sennrich. 2017. Improving word sense disambiguation in neural machine translation with sense embeddings. In *Second Conference on Machine Translation*, pages 11–19, Copenhagen, Denmark.

Jinsong Su, Deyi Xiong, Shujian Huang, Xianpei Han, and Junfeng Yao. 2015. Graph-based collective lexical selection for statistical machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1238–1247, Lisbon, Portugal.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Advances in Neural Information Processing Systems (NIPS)*, pages 3104–3112.

Haiqing Tang, Deyi Xiong, Oier Lopez de Lacalle, and Eneko Agirre. 2016. Improving translation selection with supersenses. In *Proceedings of the 26th International Conference on Computational Linguistics (COLING)*, pages 3114–3123, Osaka, Japan.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.

David Vickrey, Luke Biewald, Marc Teyssier, and Daphne Koller. 2005. Word-sense disambiguation for machine translation. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing (HLT-EMNLP)*, pages 771–778, Vancouver, BC, Canada.

Lesly Miculicich Werlen, Nikolaos Pappas, Dhananjay Ram, and Andrei Popescu-Belis. 2018. Self-attentive residual decoder for neural machine translation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 1366–1379, New Orleans, LA, USA.

Deyi Xiong and Min Zhang. 2014. A sense-based translation model for statistical machine translation. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pages 1459–1469, Baltimore MD, USA.

Zhen Yang, Wei Chen, Feng Wang, and Bo Xu. 2017. Multi-sense based neural machine translation. In *International Joint Conference on Neural Networks*, pages 3491–3497, Anchorage, AK, USA.

Biao Zhang, Deyi Xiong, Jinsong Su, and Hong Duan. 2017. A context-aware recurrent encoder for neural machine translation. *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, pages 2424–2432.