Zero-shot Word Sense Disambiguation using Sense Definition Embeddings

Sawan Kumar[1] **Sharmistha Jat**[1] **Karan Saxena**[2,*] **Partha Talukdar**[1]
[1] Indian Institute of Science, Bangalore
[2] Carnegie Mellon University, Pittsburgh
{sawankumar,sharmisthaj,ppt}@iisc.ac.in, karansax@cs.cmu.edu

Abstract

Word Sense Disambiguation (WSD) is a long-standing but open problem in Natural Language Processing (NLP). WSD corpora are typically small in size, owing to an expensive annotation process. Current supervised WSD methods treat senses as discrete labels and also resort to predicting the Most-Frequent-Sense (MFS) for words unseen during training. This leads to poor performance on rare and unseen senses. To overcome this challenge, we propose Extended WSD Incorporating Sense Embeddings (EWISE), a supervised model to perform WSD by predicting over a continuous sense embedding space as opposed to a discrete label space. This allows EWISE to generalize over both seen and unseen senses, thus achieving generalized zero-shot learning. To obtain target sense embeddings, EWISE utilizes sense definitions. EWISE learns a novel sentence encoder for sense definitions by using WordNet relations and also ConvE, a recently proposed knowledge graph embedding method. We also compare EWISE against other sentence encoders pretrained on large corpora to generate definition embeddings. EWISE achieves new state-of-the-art WSD performance.

1 Introduction

Word Sense Disambiguation (WSD) is an important task in Natural Language Processing (NLP) (Navigli, 2009). The task is to associate a word in text to its correct sense, where the set of possible senses for the word is assumed to be known a priori. Consider the noun "tie" and the following examples of its usage (Miller, 1995).

- "*he wore a vest and tie*"
- "*their record was 3 wins, 6 losses and a tie*"

It is clear that the implied sense of the word "tie" is very different in the two cases. The word is associated with *"neckwear consisting of a long narrow piece of material"* in the first example, and with *"the finish of a contest in which the winner is undecided"* in the second. The goal of WSD is to predict the right sense, given a word and its context.

WSD has been shown to be useful for popular NLP tasks such as machine translation (Neale et al., 2016; Pu et al., 2018), information extraction (Zhong and Ng, 2012; Delli Bovi et al., 2015) and question answering (Ramakrishnan et al., 2003). The task of WSD can also be viewed as an intrinsic evaluation benchmark for the semantics learned by sentence comprehension models. WSD remains an open problem despite a long history of research. In this work, we study the all-words WSD task, where the goal is to disambiguate all ambiguous words in a corpus.

Supervised (Zhong and Ng, 2010; Iacobacci et al., 2016; Melamud et al., 2016) and semi-supervised approaches (Taghipour and Ng, 2015; Yuan et al., 2016) to WSD treat the target senses as discrete labels. Treating senses as discrete labels limits the generalization capability of these models for senses which occur infrequently in the training data. Further, for disambiguation of words not seen during training, these methods fall back on using a Most-Frequent-Sense (MFS) strategy, obtained from an external resource such as WordNet (Miller, 1995). To address these concerns, unsupervised knowledge-based (KB) approaches have been introduced, which rely solely on lexical resources (e.g., WordNet). KB methods include approaches based on context-definition overlap (Lesk, 1986; Basile et al., 2014), or on the structural properties of the lexical resource (Moro et al., 2014; Weissenborn et al., 2015; Chaplot et al., 2015; Chaplot and Salakhutdinov, 2018;

[*] Work done as a Research Assistant at Indian Institute of Science, Bangalore.

Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 5670–5681
Florence, Italy, July 28 - August 2, 2019. ©2019 Association for Computational Linguistics

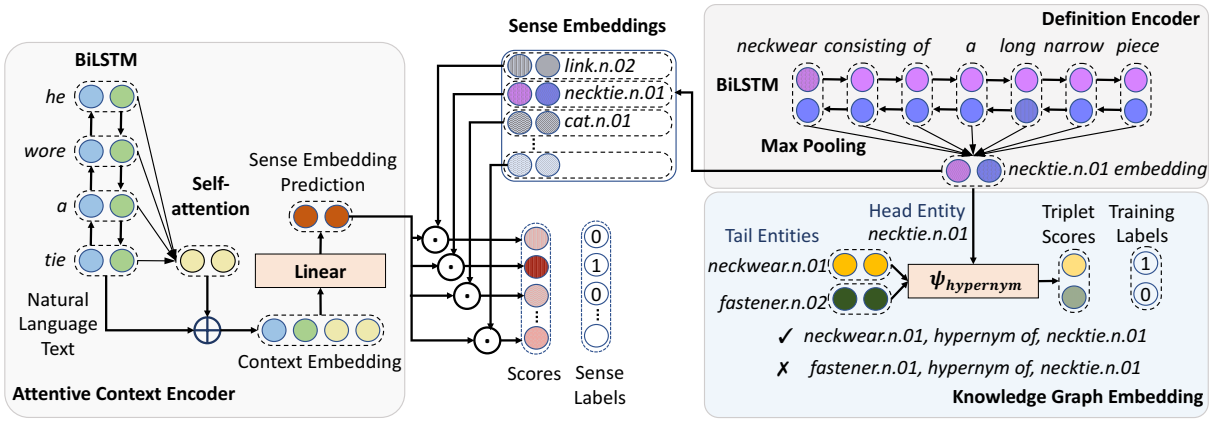

Figure 1: *Overview of WSD in EWISE*: A sequence of input tokens is encoded into context-aware embeddings using a BiLSTM and a self-attention layer ($\oplus$ indicates concatenation). The context-aware embeddings are then projected on to the space of sense embeddings. The score for each sense in the sense inventory is obtained using a dot product (indicated by $\odot$) of the sense embedding with the projected word embedding. Please see Section 4.2 for details on the context encoding and training of the context encoder. The sense embedding for each sense in the inventory is generated using a BiLSTM-Max definition encoder. The encoder is learnt using the training signal present in WordNet Graph. An example signal with hypernym relation is depicted. Please see Section 4.3 for details on learning sense embeddings.

Tripodi and Pelillo, 2017).

While knowledge-based approaches offer a way to disambiguate rare and unseen words into potentially rare senses, supervised methods consistently outperform these methods in the general setting where inference is to be carried over both frequently occurring and rare words. Recently, Raganato et al. (2017b) posed WSD as a neural sequence labeling task, further improving the state-of-the-art. Yet, owing to an expensive annotation process (Lopez de Lacalle and Agirre, 2015), there is a scarcity of sense-annotated data thereby limiting the generalization ability of supervised methods. While there has been recent interest in incorporating definitions (glosses) to overcome the supervision bottleneck for WSD (Luo et al., 2018b,a), these methods are still limited due to their treatment of senses as discrete labels.

Our hypothesis is that supervised methods can leverage lexical resources to improve on WSD for both observed and unobserved words and senses. We propose **Extended WSD Incorporating Sense Embeddings** (**EWISE**). Instead of learning a model to choose between discrete labels, EWISE learns a continuous space of sense embeddings as target. This enables generalized zero-shot learning, i.e., the ability to recognize instances of seen as well as unseen senses. EWISE utilizes sense definitions and additional information from lexical resources. We believe that natural language information manually encoded into definitions contains a rich source of information for representation learning of senses.

To obtain definition embeddings, we propose a novel learning framework which leverages recently successful Knowledge Graph (KG) embedding methods (Bordes et al., 2013; Dettmers et al., 2018). We also compare against sentence encoders pretrained on large corpora.

In summary, we make the following contributions in this work.

- We propose EWISE, a principled framework to learn from a combination of sense-annotated data, dictionary definitions and lexical knowledge bases.
- We propose the use of sense embeddings instead of discrete labels as the targets for supervised WSD, enabling generalized zero-shot learning.
- Through extensive evaluation, we demonstrate the effectiveness of EWISE over state-of-the-art baselines.

EWISE source code is available at `https://github.com/malllabiisc/EWISE`

2 Related Work

Classical approaches to supervised WSD relied on extracting potentially relevant features and learning classifiers independently for each word

(Zhong and Ng, 2010). Extensions to use distributional word representations have been proposed (Iacobacci et al., 2016). Semi-supervised approaches learn context representations from unlabeled data, followed by a nearest neighbour classification (Melamud et al., 2016) or label propagation (Yuan et al., 2016). Recently, Raganato et al. (2017b) introduced neural sequence models for joint disambiguation of words in a sentence. All of these methods rely on sense-annotated data and, optionally, additional unlabeled corpora.

Lexical resources provide an important source of knowledge about words and their meanings. Recent work has shown that neural networks can extract semantic information from dictionary definitions (Bahdanau et al., 2017; Bosc and Vincent, 2018). In this work, we use dictionary definitions to get representations of word meanings.

Dictionary definitions have been used for WSD, motivated by the classical method of Lesk (Lesk, 1986). The original as well as subsequent modifications of the algorithm (Banerjee and Pedersen, 2003), including using word embeddings (Basile et al., 2014), operate on the hypothesis that the definition of the correct sense has a high overlap with the context in which a word is used. These methods tend to rely on heuristics based on insights about natural language text and their definitions. More recently, gloss (definition)-augmented neural approaches have been proposed which integrate a module to score definition-context similarity (Luo et al., 2018b,a), and achieve state-of-the-art results. We differ from these works in that we use the embeddings of definitions as the target space of a neural model, while learning in a supervised setup. Also, we don't rely on any overlap heuristics, and use a single definition for a given sense as provided by WordNet.

One approach for obtaining continuous representations for definitions is to use **Universal Sentence Representations**, which have been explored to allow transfer learning from large unlabeled as well as labeled data (Conneau et al., 2017; Cer et al., 2018). There has also been interest in learning deep contextualized word representations (Peters et al., 2018; Devlin et al., 2019). In this work, we evaluate definition embeddings obtained using these methods.

Structural Knowledge available in lexical resources such as WordNet has motivated several unsupervised knowledge-based approaches for WSD. Graph based techniques have been used to match words to the most relevant sense (Navigli and Lapata, 2010; Sinha and Mihalcea, 2007; Agirre et al., 2014; Moro et al., 2014; Chaplot and Salakhutdinov, 2018).

Our work differs from these methods in that we use structural knowledge to learn better representations of definitions, which are then used as targets for the WSD model. To learn a meaningful encoder for definitions we rely on knowledge graph embedding methods, where we represent an entity by the encoding of its definition. TransE (Bordes et al., 2013) models relations between entities as translations operating on the embeddings of the corresponding entities. ConvE (Dettmers et al., 2018), a more recent method, utilizes a multi-layer convolutional network, allowing it to learn more expressive features.

Predicting in an embedding space is key to our methods, allowing generalized zero shot learning capability, as well as incorporating definitions and structural knowledge. The idea has been explored in the context of zero-shot learning (Xian et al., 2018). Tying the input and output embeddings of language models (Press and Wolf, 2017) resembles our approach.

3 Background

In this work, we propose to use the training signal present in WordNet relations to learn encoders for definitions (Section 4.3.2). To learn from WordNet relations, we employ recently popular Knowledge Graph (KG) Embedding learning methods. In Section 3.1, we briefly introduce the framework for KG Embedding learning, and present the specific formulations for TransE and ConvE.

3.1 Knowledge Graph Embeddings

Knowledge Graphs, a set of relations defined over a set of entities, provide an important field of research for representation learning. Methods for learning representations for both entities and relations have been explored (Wang et al., 2017) with an aim to represent graphical knowledge. Of particular significance is the task of link prediction, i.e., predicting missing links (edges) in the graph.

A Knowledge Graph is typically comprised of a set K of N triples (h, l, t), where head h and tail t are entities, and l denotes a relation.

TransE defines a scoring function for a triple (h, l, t), as the dissimilarity between the head em-

bedding, translated by the relation embedding, and the tail embedding:

$$d_{h,l,t} = ||e_h + e_l - e_t||_2^2, \quad (1)$$

where, e_h, e_t and e_l are parameters to be learnt.

A margin based criterion, with margin γ, can then be formulated as:

$$L_T = \sum_{(h,l,t)\in K} \sum_{(h',l,t')\in K'} [\gamma + d_{h,l,t} - d_{h',l,t'}]_+, \quad (2)$$

where K' is a set of corrupted triples (Bordes et al., 2013), and $[x]_+$ refers to the positive part of x.

ConvE formulates the scoring function $\psi_l(e_h, e_t)$ for a triple (h, l, t) as:

$$\psi_l(e_h, e_t) = f(\text{vec}(f([\overline{e_h}; \overline{e_l}] * w))W)e_t, \quad (3)$$

where e_h and e_t are entity parameters, e_l is a relation parameter, $\overline{x}$ denotes a 2D reshaping of x, w denotes the filters for 2D convolution, $\text{vec}(x)$ denotes the vectorization of x, W represents a linear transformation, and f denotes a rectified linear unit.

For a given head entity h, the score $\psi_l(e_h, e_t)$ is computed with each entity in the graph as a tail. Probability estimates for the validity of a triple are obtained by applying a logistic sigmoid function to the scores:

$$p = \sigma(\psi_l(e_h, e_t)). \quad (4)$$

The model is then trained using a binary cross entropy loss:

$$L_C = -\frac{1}{N}\sum_i (t_i.log(p_i) + (1 - t_i).log(1 - p_i)), \quad (5)$$

where t_i is 1 when $(h, l, t) \in K$ and 0, otherwise.

4 EWISE

EWISE is a general WSD framework for learning from sense-annotated data, dictionary definitions and lexical knowledge bases (Figure 1).

EWISE addresses a key issue with existing supervised WSD systems. Existing systems use discrete sense labels as targets for WSD. This limits the generalization capability to only the set of annotated words in the corpus, with reliable learning only for the word-senses which occur with high relative frequency. In this work, we propose using continuous space embeddings of senses as targets for WSD, to overcome the aforementioned supervision bottleneck.

To ensure generalized zero-shot learning capability, it is important that the target sense embeddings be obtained independent of the WSD task learning. We use definitions of senses available in WordNet to obtain sense embeddings. Using Dictionary Definitions to obtain the representation for a sense enables us to benefit from the semantic overlap between definitions of different senses, while also providing a natural way to handle unseen senses.

In Section 4.1, we state the task of WSD formally. We then describe the components of EWISE in detail. Here, we briefly discuss the components:

- **Attentive Context Encoder**: EWISE uses a Bi-directional LSTM (BiLSTM) encoder to convert the sequence of tokens in the input sentence into context-aware embeddings. Self-attention is used to enhance the context for disambiguating the current word, followed by a projection layer to produce sense embeddings for each input token. The architecture is detailed in Section 4.2.

- **Definition Encoder**: In EWISE, definition embeddings are learnt independent of the WSD task. In Section 4.3.1, we detail the usage of pretrained sentence encoders as baseline models for encoding definitions. In Section 4.3.2, we detail our proposed method to learn an encoder for definitions using structural knowledge in WordNet.

4.1 The WSD Task

WSD is a classification problem for a word w (e.g., bank) in a context c, with class labels being the word senses (e.g., financial institution).

We consider the all-words WSD task, where all content words - nouns, verbs, adjectives, adverbs - need to be disambiguated (Raganato et al., 2017a). The set of all possible senses for a word is given by a predefined sense inventory, such as WordNet. In this work, we use sense candidates as provided in the evaluation framework of (Raganato et al., 2017a) which has been created using WordNet.

More precisely, given a variable-length sequence of words $x =< x^1 \dots x^T >$, we need to predict a sequence of word senses $y =<$

$y^1 \ldots y^T >$. Output word sense y^i comes from a predefined sense inventory S. During inference, the set of candidate senses S_w for input word w is assumed to be known a priori.

4.2 Attentive Context Encoder

In this section, we detail how EWISE encodes the context of a word to be disambiguated using BiLSTMs (Hochreiter and Schmidhuber, 1997). BiLSTMs have been shown to be successful for generating effective context dependent representations for words. Following Raganato et al. (2017b), we use a BiLSTM with a self-attention layer to obtain sense-aware context specific representations of words. The sense embedding for a word is obtained through a projection of the context embedding. We then train the model with independently trained sense embeddings (Section 4.3) as target embeddings.

Our model architecture is shown in Figure 1. The model processes a sequence of tokens $x^i, i \in [T]$ in a given sentence input by first representing each token with a real-valued vector representation, e^i, via an embedding matrix $W_e \in R^{|V|*d}$, where V is the vocabulary size and d is the size of the embeddings. The vector representations are then input to a 2 layer bidirectional LSTM encoder. Each word is represented by concatenating the forward h^i_f and backward h^i_b hidden state vectors of the second LSTM layer.

$$u^i = [h^i_f, h^i_b] \tag{6}$$

Following Vaswani et al. (2017), we use a scaled dot-product attention mechanism to get context information at each timestep t. Attention queries, keys and values are obtained using projection matrices W_q, W_k and W_v respectively, while the size of the projected key (d_k) is used to scale the dot-product between queries and values.

$$\begin{aligned} e^i_t &= \text{dot}(W_q u^i, W_k u^t); t \in [1, T] \\ a^i &= \text{softmax}(\frac{e^i}{\sqrt{d_k}}) \\ c^i &= \sum_{t \in [1,T]} a^i_t . W_v u^t \\ r^i &= [u^i, c^i] \end{aligned} \tag{7}$$

A projection layer (fully connected linear layer) maps this context-aware word representation r_i to v_i in the space of sense embeddings.

$$v^i = W_l r^i \tag{8}$$

During training, we multiply this with the sense embeddings of all senses in the inventory, to obtain a score for each output sense. A bias term is added to this score, where the bias is obtained as the dot product between the sense embedding and a learned parameter b. A softmax layer then generates probability estimates for each output sense.

$$\hat{p}^i_j = \text{softmax}(\text{dot}(v^i, \rho_j) + \text{dot}(b, \rho_j)); \quad \rho_j \in S \tag{9}$$

The cross entropy loss for annotated word x^i is given by:

$$L^i_{\text{wsd}} = -\sum_j (z^i_j \log(\hat{p}^i_j)), \tag{10}$$

where z^i is the one-hot representation of the target sense y^i in the sense inventory S. The network parameters are learnt by minimizing the average cross entropy loss over all annotated words in a batch.

During inference, for each word x^i, we select the candidate sense with the highest score.

$$\hat{y}^i = \text{argmax}_j(\text{dot}(v^i, \rho_j) + \text{dot}(b, \rho_j)); \quad \rho_j \in S_{x^i} \tag{11}$$

4.3 Definition Encoder

In this section, we detail how target sense embeddings are obtained in EWISE.

4.3.1 Pretrained Sentence Encoders

We use pretrained sentence representation models, InferSent (Conneau et al., 2017) and USE (Cer et al., 2018) to encode definitions, producing sense embeddings of sizes 4096 and 512, respectively.

We also experiment with deep context encoders, ELMO (Peters et al., 2018) and BERT (Devlin et al., 2019) to obtain embeddings for definitions. In each case, we encode a definition using the available pretrained models, producing a context embedding for each word in the definition. A fixed length representation is then obtained by averaging over the context embeddings of the words in the definition, from the final layer. This produces sense embeddings of sizes 1024 with both ELMO and BERT.

4.3.2 Knowledge Graph Embedding

WordNet contains a knowledge graph, where the entities of the graph are senses (synsets), and re-

	Dev	Test Datasets				Concatenation of All Test Datasets				
	SE7	SE2	SE3	SE13	SE15	Nouns	Verbs	Adj.	Adv.	ALL
WordNet S1	55.2	66.8	66.2	63.0	67.8	67.6	50.3	74.3	80.9	65.2
Non-neural baselines										
MFS (Using training data)	54.5	65.6	66.0	63.8	67.1	67.7	49.8	73.1	80.5	65.5
IMS+emb (2016)ˆ	62.6	72.2	70.4	65.9	71.5	71.9	56.6	75.9	84.7	70.1
Lesk_{ext}+emb (2014)*	56.7	63.0	63.7	66.2	64.6	70.0	51.1	51.7	80.6	64.2
$\text{UKB}_{\text{gloss}}$+w2w (2014)*	42.9	63.5	55.4	62.9	63.3	64.9	41.4	69.5	69.7	61.1
Babelfy (2014)	51.6	67.0	63.5	66.4	70.3	68.9	50.7	73.2	79.8	66.4
Context2Vec (2016) ˆ	61.3	71.8	69.1	65.6	71.9	71.2	57.4	75.2	82.7	69.6
WSD-TM (2018)	55.6	69.0	66.9	65.3	69.6	69.7	51.2	76.0	80.9	66.9
Neural baselines										
BiLSTM+att+LEX (2017b)	63.7	72.0	69.4	66.4	70.8	71.6	57.1	75.6	83.2	69.7
BiLSTM+att+LEX+POS (2017b)	64.8	72.0	69.1	66.9	71.5	71.5	57.5	75.0	83.8	69.9
GAS_{ext} (Linear) (2018b)*	–	72.4	70.1	67.1	72.1	71.9	58.1	76.4	84.7	70.4
GAS_{ext} (Concatenation) (2018b)*	–	72.2	70.5	67.2	72.6	72.2	57.7	76.6	**85.0**	70.6
CAN_{s} (2018a)*	–	72.2	70.2	69.1	72.2	73.5	56.5	76.6	83.3	70.9
HCAN (2018a)*	–	72.8	70.3	68.5	72.8	72.7	58.2	77.4	84.1	71.1
EWISE (ConvE)*	**67.3**	**73.8**	**71.1**	**69.4**	**74.5**	**74.0**	**60.2**	**78.0**	82.1	**71.8**

Table 1: *Comparison of F1-scores for fine-grained all-words WSD* on Senseval and SemEval datasets in the framework of Raganato et al. (2017a). The F1 scores on different POS tags (Nouns, Verbs, Adjectives, and Adverbs) are also reported. WordNet S1 and MFS provide most-frequent-sense baselines. * represents models which access definitions, while ˆ indicates models which don't access any external knowledge. EWISE (ConvE) is the proposed approach, where the ConvE method was used to generate the definition embeddings. Both the non-neural and neural supervised baselines presented here rely on a back-off mechanism, using WordNet S1 for words unseen during training. For each dataset, the highest score among existing systems with a statistically significant difference (unpaired t-test, $p < 0.05$) from EWISE is underlined. EWISE, which is capable of generalizing to unseen words and senses, doesn't use any back-off. EWISE consistently outperforms all supervised and knowledge-based systems, except for adverbs. Please see Section 6.1 for details. While the overall performance of EWISE is comparable to the neural baselines in terms of statistical significance, the value of EWISE lies in its ability to handle unseen and rare words and senses (See Section 6.3). Further, among the models compared, EWISE is the only system which is statistically significant (unpaired t-test, $p < 0.01$) with respect to the WordNet S1 baseline across all test datasets.

lations are defined over these senses. Example relations include hypernym and part_of. With each entity (sense), there is an associated text definition.

We propose to use WordNet relations as the training signal for learning definition encoders. The training set K is comprised of triples (h, l, t), where head h and tail t are senses, and l is a relation. Also, g_x denotes the definition of entity x, as provided by WordNet. The dataset contains 18 WordNet relations (Bordes et al., 2013).

The goal is to learn a sentence encoder for definitions and we select the BiLSTM-Max encoder architecture due to its recent success in sentence representation (Conneau et al., 2017). The words in the definition are encoded by a 2-layer BiLSTM to obtain context-aware embeddings for each word. A fixed length representation is then obtained by Max Pooling, i.e., selecting the maximum over each dimension. We denote this definition encoder by $q(.)$.

TransE We modify the dissimilarity measure in TransE (Equation 1) to represent both head (h) and tail (t) entities by an encoding of their definitions.

$$d_{h,l,t} = -\text{cosine}(q(h) + e_l, q(t)) \quad (12)$$

The parameters of the BiLSTM model q and the relation embeddings e_l are then learnt by minimizing the loss function in Equation 2.

ConvE We modify the scoring function of ConvE (Equation 3), to represent a head entity by the encoding of its definition.

$$\psi_l(e_h, e_t) = f(\text{vec}(f([\overline{q(h)}; \overline{e_l}] * w))W)e_t \quad (13)$$

Note that we represent only the head entity with an encoding of its definition while the tail entity t is still represented by parameter e_t. This helps restrict the size of the computation graph.

The parameters of the model q, e_l and e_t are then learnt by minimizing the binary cross-entropy loss function in Equation 5.

5 Experimental Setup

In this section, we provide details on the training and evaluation datasets. The training details are

captured in Appendix A.

5.1 Data

We use the English all-words WSD benchmarks for evaluating our models:

1. SensEval-2 (Palmer et al., 2001)
2. SensEval-3 (Snyder and Palmer, 2004)
3. SemEval-2013 (Navigli et al., 2013)
4. SemEval-2015 (Moro and Navigli, 2015)
5. ALL (Raganato et al., 2017a)

Following (Raganato et al., 2017b), we use SemEval-2007 (Pradhan et al., 2007) as our development set. We use SemCor 3.0 (Miller et al., 1993) as our training set. To enable a fair comparison, we used the dataset versions provided by (Raganato et al., 2017a). For our experiments, we used the definitions available in WordNet 3.0.

6 Evaluation

In this section, we aim to answer the following questions:

- Q1: How does EWISE compare to state-of-the-art methods on standardized test sets? (Section 6.1)
- Q2: What is the effect of ablating key components from EWISE? (Section 6.2)
- Q3: Does EWISE generalize to rare and unseen words (Section 6.3.1) and senses (Section 6.3.2)?
- Q4: Can EWISE learn with less annotated data? (Section 6.4)

6.1 Overall Results

In this section, we report the performance of EWISE on the fine-grained all-words WSD task, using the standardized benchmarks and evaluation methodology introduced in Raganato et al. (2017a). In Table 1, we report the F1 scores for EWISE, and compare against the best reported supervised and knowledge-based methods.

WordNet S1 is a strong baseline obtained by using the most frequent sense of a word as listed in WordNet. MFS is a most-frequent-sense baseline obtained through the sense frequencies in the training corpus.

Context2Vec (Melamud et al., 2016), an unsupervised model for learning generic context embeddings, enables a strong baseline for supervised WSD while using a simplistic approach (nearest-neighbour algorithm).

IMS+emb (Iacobacci et al., 2016) takes the classical approach of extracting relevant features and learning an SVM for WSD. Lesk_{ext}+emb (Basile et al., 2014) relies on definition-context overlap heuristics. $\text{UKB}_{\text{gloss}}$w2w (Agirre et al., 2014), Babelfy (Moro et al., 2014) and WSD-TM (Chaplot and Salakhutdinov, 2018) provide unsupervised knowledge-based methods. Among neural baselines, we compare against the neural sequence modeling approach in BiLSTM+att+LEX(+POS) (Raganato et al., 2017b). GAS (Luo et al., 2018b) and HCAN (Luo et al., 2018a) are recent neural models which exploit sense definitions. EWISE consistently outperforms all supervised and knowledge-based methods, improving upon the state-of-the-art by 0.7 point in F1 on the ALL dataset. Further, EWISE improves WSD performance across all POS tags (Table 1) except adverbs.

Back-off : Traditional supervised approaches can't handle unseen words. WordNet S1 is used as a back-off strategy for words unseen during training. EWISE is capable of generalizing to unseen words and senses and doesn't use any back-off.

6.2 Ablation Study for EWISE

Ablation on ALL dataset	
EWISE (ConvE)	**71.8**
- w/o Sense embeddings (with back-off)	69.3
- w/o Sense embeddings (w/o back-off)	61.8
WordNet S1	65.2

Table 2: *Ablation study for EWISE* (ConvE) on the ALL dataset. Removal of sense embeddings (rows 2 and 3) results in significant performance degradation, establishing their importance in WSD. Please see Section 6.2 for details.

We provide an ablation study of EWISE on the ALL dataset in Table 2. To investigate the effect of using definition embeddings in EWISE, we trained a BiLSTM model without any externally obtained sense embeddings. This model can make predictions only on words seen during training, and is evaluated with or without a back-off strategy (WordNet S1) for unseen words (row 2 and 3). The results demonstrate that incorporating sense

embeddings is key to EWISE's performance. Further, the generalization capability of EWISE is illustrated by the improvement in F1 in the absence of a back-off strategy (10.0 points).

	Test Datasets				
	SE2	SE3	SE13	SE15	ALL
USE	73.0	70.6	**70.9**	73.7	71.5
InferSent	72.7	70.2	69.9	73.7	71.2
ELMO	72.5	70.7	68.6	72.6	70.8
BERT	73.0	69.7	70.0	73.7	71.2
DeConf	71.3	67.0	67.9	73.0	69.3
TransE	72.8	**71.4**	70.5	73.1	71.6
ConvE	**73.8**	71.1	69.4	**74.5**	**71.8**

Table 3: *Comparison of F1 scores with different sense embeddings as targets for EWISE.* While pre-trained embedding methods (USE, InferSent, ELMO, BERT) and DeConf provide impressive results, the KG embedding methods (TransE and ConvE) perform competitively or better by learning to encode definitions using WordNet alone. Please see Section 6.2 for details.

Next, we investigate the impact of the choice of sense embeddings used as the target for EWISE (Table 3), on the ALL dataset. We compare definition embeddings learnt using structural knowledge (TransE, ConvE; See Section 4.3.2) against definition embeddings obtained from pre-trained sentence and context encoders (USE, InferSent, ELMO, BERT; See Section 4.3.1). We also compared with off-the-shelf sense embeddings (DeConf) (Pilehvar and Collier, 2016), where definitions are not used. The results justify the choice of learning definition embeddings to represent senses.

6.3 Detailed Results

We provide detailed results for EWISE on the ALL dataset, compared against BiLSTM-A (BiLSTM+attention) baseline which is trained to predict in the discrete label space (Raganato et al., 2017b). We also compare against WordNet S1 and knowledge-based methods, Lesk$_{ext}$+emb and Babelfy, available in the evaluation framework of Raganato et al. (2017a).

6.3.1 WSD on Rare Words

In this section, we investigate a key claim of EWISE - the ability to disambiguate unseen and rare words. We evaluate WSD models based on different frequencies of annotated words in the training set in Figure 2. EWISE outperforms the supervised as well as knowledge-based baselines for rare as well as frequent words. The bar plot

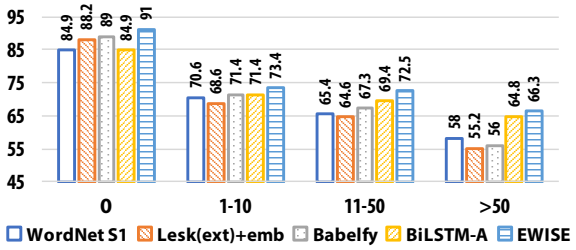

Figure 2: *Comparison of F1 scores for different frequencies of annotated words in the train set.* EWISE provides significant gains for unseen, rare as well as frequently observed annotated words. Please see Section 6.3.1 for details.

on the left (frequency=0) indicates the zero-shot learning capability of EWISE. While traditional supervised systems are limited to WordNet S1 performance (by using it as back-off for words with no annotations in the training set), EWISE provides a significant boost over both WordNet S1 as well as knowledge-based systems.

6.3.2 WSD on Rare Senses

	MFS	LFS
WordNet S1	100.0	0.0
Lesk(ext)+emb	92.7	9.4
Babelfy	93.9	12.2
BiLSTM-A	93.4	22.9
EWISE	93.5	**31.2**

Table 4: *Comparison of F1 scores on different sense frequencies.* EWISE outperforms baselines on infrequent senses, without sacrificing the performance on the most frequent sense examples. Please see Section 6.3.2 for details.

To investigate the ability to generalize to rare senses, we partition the ALL test set into two parts - the set of instances labeled with the most frequent sense of the corresponding word (MFS), and the set of remaining instances (LFS: Least Frequent Senses). Postma et al. (2016) note that existing methods learn well on the MFS set, while doing poorly ($\sim 20\%$) on the LFS set.

In Table 4, we evaluate the performance of EWISE and baseline models on MFS and LFS sets. We note that EWISE provides significant gains over a neural baseline (BiLSTM-A), as well as knowledge based methods on the LFS set, while maintaining high accuracy on the MFS set. The gain obtained on the LFS set is consistent with our hypothesis that predicting over sense embeddings enables generalization to rare senses.

6.4 Size of Training Data

	Size of training data	F1	
		Without back-off	With back-off
WordNet S1			65.2
EWISE	20%	66.8	67.0
	50%	70.1	69.2
	100%	71.8	71.0

Table 5: *Performance of EWISE with varying sizes of training data.* With only 20% of training data, EWISE is able to outperform the most-frequent-sense baseline of WordNet S1. Please see Section 6.4 for details.

In this section, we investigate if EWISE can learn efficiently from less training data, given its increased supervision bandwidth (sense embeddings instead of sense labels). In Table 5, we report the performance of EWISE on the ALL dataset with varying sizes of the training data. We note that with only 50% of training data, EWISE already competes with several supervised approaches (Table 1), while with just 20% of training data, EWISE is able to outperform the strong WordNet S1 baseline. For reference, we also present the performance of EWISE when we use back-off (WordNet S1) for words unseen during training.

7 Conclusion and Future Work

We have introduced EWISE, a general framework for learning WSD from a combination of sense-annotated data, dictionary definitions and Lexical Knowledge Bases. EWISE uses sense embeddings as targets instead of discrete sense labels. This helps the model gain zero-shot learning capabilities, demonstrated through ablation and detailed analysis. EWISE improves state-of-the-art results on standardized benchmarks for WSD. We are releasing EWISE code to promote reproducible research.

This paper should serve as a starting point to better investigate WSD on out-of-vocabulary words. Our modular architecture opens up various avenues for improvements in few-shot learning for WSD, viz., context encoder, definition encoder, and leveraging structural knowledge. Another potential future work would be to explore other ways of providing rich supervision from textual descriptions as targets.

Acknowledgments

We thank the anonymous reviewers for their constructive comments. This work is supported in part by the Ministry of Human Resource Development (Government of India), and by a travel grant from Microsoft Research India.

References

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

Dzmitry Bahdanau, Tom Bosc, Stanisaw Jastrzebski, Edward Grefenstette, Pascal Vincent, and Yoshua Bengio. 2017. Learning to compute word embeddings on the fly. *arXiv preprint arXiv:1706.00286*.

Satanjeev Banerjee and Ted Pedersen. 2003. Extended gloss overlaps as a measure of semantic relatedness. In *Ijcai*, volume 3, pages 805–810.

Pierpaolo Basile, Annalina Caputo, and Giovanni Semeraro. 2014. An enhanced Lesk word sense disambiguation algorithm through a distributional semantic model. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 1591–1600, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *Advances in neural information processing systems*, pages 2787–2795.

Tom Bosc and Pascal Vincent. 2018. Auto-encoding dictionary definitions into consistent word embeddings. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1522–1532, Brussels, Belgium. Association for Computational Linguistics.

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Brian Strope, and Ray Kurzweil. 2018. Universal sentence encoder for English. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 169–174, Brussels, Belgium. Association for Computational Linguistics.

Devendra Singh Chaplot, Pushpak Bhattacharyya, and Ashwin Paranjape. 2015. Unsupervised word sense disambiguation using markov random field and dependency parser. In *AAAI*, pages 2217–2223.

Devendra Singh Chaplot and Ruslan Salakhutdinov. 2018. Knowledge-based word sense disambiguation using topic models. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Alexis Conneau, Douwe Kiela, Holger Schwenk, Loïc Barrault, and Antoine Bordes. 2017. Supervised learning of universal sentence representations from natural language inference data. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 670–680, Copenhagen, Denmark. Association for Computational Linguistics.

Claudio Delli Bovi, Luis Espinosa-Anke, and Roberto Navigli. 2015. Knowledge base unification via sense embeddings and disambiguation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 726–736, Lisbon, Portugal. Association for Computational Linguistics.

Tim Dettmers, Pasquale Minervini, Pontus Stenetorp, and Sebastian Riedel. 2018. Convolutional 2d knowledge graph embeddings. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Comput.*, 9(8):1735–1780.

Ignacio Iacobacci, Mohammad Taher Pilehvar, and Roberto Navigli. 2016. Embeddings for word sense disambiguation: An evaluation study. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 897–907, Berlin, Germany. Association for Computational Linguistics.

Oier Lopez de Lacalle and Eneko Agirre. 2015. A methodology for word sense disambiguation at 90% based on large-scale CrowdSourcing. In *Proceedings of the Fourth Joint Conference on Lexical and Computational Semantics*, pages 61–70, Denver, Colorado. Association for Computational Linguistics.

Michael Lesk. 1986. Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone. In *Proceedings of the 5th annual international conference on Systems documentation*, pages 24–26. ACM.

Fuli Luo, Tianyu Liu, Zexue He, Qiaolin Xia, Zhifang Sui, and Baobao Chang. 2018a. Leveraging gloss knowledge in neural word sense disambiguation by hierarchical co-attention. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1402–1411, Brussels, Belgium. Association for Computational Linguistics.

Fuli Luo, Tianyu Liu, Qiaolin Xia, Baobao Chang, and Zhifang Sui. 2018b. Incorporating glosses into neural word sense disambiguation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2473–2482, Melbourne, Australia. Association for Computational Linguistics.

Oren Melamud, Jacob Goldberger, and Ido Dagan. 2016. context2vec: Learning generic context embedding with bidirectional LSTM. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 51–61, Berlin, Germany. Association for Computational Linguistics.

George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

George A. Miller, Claudia Leacock, Randee Tengi, and Ross T. Bunker. 1993. A semantic concordance. In *HUMAN LANGUAGE TECHNOLOGY: Proceedings of a Workshop Held at Plainsboro, New Jersey, March 21-24, 1993*.

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.

Andrea Moro, Alessandro Raganato, and Roberto Navigli. 2014. Entity linking meets word sense disambiguation: a unified approach. *Transactions of the Association for Computational Linguistics*, 2:231–244.

Roberto Navigli. 2009. Word sense disambiguation: A survey. *ACM Computing Surveys (CSUR)*, 41(2):10.

Roberto Navigli, David Jurgens, and Daniele Vannella. 2013. SemEval-2013 task 12: Multilingual word sense disambiguation. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 222–231, Atlanta, Georgia, USA. Association for Computational Linguistics.

Roberto Navigli and Mirella Lapata. 2010. An experimental study of graph connectivity for unsupervised word sense disambiguation. *IEEE transactions on pattern analysis and machine intelligence*, 32(4):678–692.

Steven Neale, Luís 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 Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, pages 2777–2783, Portorož, Slovenia. European Language Resources Association (ELRA).

Martha Palmer, Christiane Fellbaum, Scott Cotton, Lauren Delfs, and Hoa Trang Dang. 2001. English tasks: All-words and verb lexical sample. In *Proceedings of SENSEVAL-2 Second International Workshop on Evaluating Word Sense Disambiguation Systems*, pages 21–24, Toulouse, France. Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Mohammad Taher Pilehvar and Nigel Collier. 2016. De-conflated semantic representations. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1680–1690, Austin, Texas. Association for Computational Linguistics.

Marten Postma, Ruben Izquierdo Bevia, and Piek Vossen. 2016. More is not always better: balancing sense distributions for all-words word sense disambiguation. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3496–3506, Osaka, Japan. The COLING 2016 Organizing Committee.

Sameer Pradhan, Edward Loper, Dmitriy Dligach, and Martha Palmer. 2007. SemEval-2007 task-17: English lexical sample, SRL and all words. In *Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007)*, pages 87–92, Prague, Czech Republic. Association for Computational Linguistics.

Ofir Press and Lior Wolf. 2017. Using the output embedding to improve language models. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 157–163, Valencia, Spain. Association for Computational Linguistics.

Xiao Pu, Nikolaos Pappas, James Henderson, and Andrei Popescu-Belis. 2018. Integrating weakly supervised word sense disambiguation into neural machine translation. *Transactions of the Association for Computational Linguistics*, 6:635–649.

Alessandro Raganato, Jose Camacho-Collados, and Roberto Navigli. 2017a. Word sense disambiguation: A unified evaluation framework and empirical comparison. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 99–110, Valencia, Spain. Association for Computational Linguistics.

Alessandro Raganato, Claudio Delli Bovi, and Roberto Navigli. 2017b. Neural sequence learning models for word sense disambiguation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1156–1167, Copenhagen, Denmark. Association for Computational Linguistics.

Ganesh Ramakrishnan, Apurva Jadhav, Ashutosh Joshi, Soumen Chakrabarti, and Pushpak Bhattacharyya. 2003. Question answering via Bayesian inference on lexical relations. In *Proceedings of the ACL 2003 Workshop on Multilingual Summarization and Question Answering*, pages 1–10, Sapporo, Japan. Association for Computational Linguistics.

Ravi Sinha and Rada Mihalcea. 2007. Unsupervised graph-basedword sense disambiguation using measures of word semantic similarity. In *Semantic Computing, 2007. ICSC 2007. International Conference on*, pages 363–369. IEEE.

Benjamin Snyder and Martha Palmer. 2004. The English all-words task. In *Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text*, pages 41–43, Barcelona, Spain. Association for Computational Linguistics.

Kaveh Taghipour and Hwee Tou Ng. 2015. Semi-supervised word sense disambiguation using word embeddings in general and specific domains. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 314–323, Denver, Colorado. Association for Computational Linguistics.

Rocco Tripodi and Marcello Pelillo. 2017. A game-theoretic approach to word sense disambiguation. *Computational Linguistics*, 43(1):31–70.

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.

Quan Wang, Zhendong Mao, Bin Wang, and Li Guo. 2017. Knowledge graph embedding: A survey of approaches and applications. *IEEE Transactions*

on Knowledge and Data Engineering, 29(12):2724–2743.

Dirk Weissenborn, Leonhard Hennig, Feiyu Xu, and Hans Uszkoreit. 2015. Multi-objective optimization for the joint disambiguation of nouns and named entities. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 596–605, Beijing, China. Association for Computational Linguistics.

Yongqin Xian, Christoph H Lampert, Bernt Schiele, and Zeynep Akata. 2018. Zero-shot learning-a comprehensive evaluation of the good, the bad and the ugly. *IEEE transactions on pattern analysis and machine intelligence*.

Dayu Yuan, Julian Richardson, Ryan Doherty, Colin Evans, and Eric Altendorf. 2016. Semi-supervised word sense disambiguation with neural models. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 1374–1385, Osaka, Japan. The COLING 2016 Organizing Committee.

Zhi Zhong and Hwee Tou Ng. 2010. It makes sense: A wide-coverage word sense disambiguation system for free text. In *Proceedings of the ACL 2010 System Demonstrations*, pages 78–83, Uppsala, Sweden. Association for Computational Linguistics.

Zhi Zhong and Hwee Tou Ng. 2012. Word sense disambiguation improves information retrieval. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 273–282, Jeju Island, Korea. Association for Computational Linguistics.

A Training Details

For both context and definition encoding, we used BiLSTMs of hidden size 2048. The input embeddings for the BiLSTM was initialized with GloVe[1] (Pennington et al., 2014) embeddings and kept fixed during training. We used the Adam optimizer for learning all our models.

WSD: We used an initial learning rate of 0.0001, a batch size of 32, and trained our models for a maximum of 200 epochs. For each run, we select the model with the best F1 score on the development set (SemEval-2007).

During training, we consider the entire sense inventory (the global pool of candidate senses of all words) for learning. During inference, for fair comparison with baselines, we disambiguate between candidates senses of a word as provided in WordNet.

TransE: We use training data from Bordes et al. (2013)[2]. We used an initial learning rate of 0.001, a batch size of 32, and trained for a maximum of 1000 epochs. The embedding size was fixed to 4096.

ConvE: We use the learning framework of Dettmers et al. (2018), and learned the model with an inital learning rate of 0.0001, a batch size of 128, label smoothing of 0.1, and a maximum of 500 epochs. We found that the best results were obtained by pretraining the entity and relation embedding using Equation 3 and then training the definition encoder using Equation 13 while allowing all parameters to train. The embedding size was fixed to 4096.

[1] http://nlp.stanford.edu/data/glove.840B.300d.zip

[2] https://everest.hds.utc.fr/lib/exe/fetch.php?media=en:wordnet-mlj12.tar.gz