Connect-the-Dots: Bridging Semantics between Words and Definitions via Aligning Word Sense Inventories

Wenlin Yao Xiaoman Pan Lifeng Jin Jianshu Chen Dian Yu Dong Yu
Tencent AI Lab, Bellevue, WA, USA
{wenlinyao,xiaomanpan,lifengjin,jianshuchen,yudian,dyu}@tencent.com

Abstract

Word Sense Disambiguation (WSD) aims to automatically identify the exact meaning of one word according to its context. Existing supervised models struggle to make correct predictions on rare word senses due to limited training data and can only select the best definition sentence from one predefined word sense inventory (e.g., WordNet). To address the data sparsity problem and generalize the model to be independent of one predefined inventory, we propose a gloss alignment algorithm that can align definition sentences (glosses) with the same meaning from different sense inventories to collect rich lexical knowledge. We then train a model to identify semantic equivalence between a target word in context and one of its glosses using these aligned inventories, which exhibits strong transfer capability to many WSD tasks[1]. Experiments on benchmark datasets show that the proposed method improves predictions on both frequent and rare word senses, outperforming prior work by 1.2% on the All-Words WSD Task and 4.3% on the Low-Shot WSD Task. Evaluation on WiC Task also indicates that our method can better capture word meanings in context.

1 Introduction

Human language is inherently ambiguous since words can have various meanings in different contexts. Word Sense Disambiguation (WSD) aims to automatically identify the correct sense (meaning) of the target word within a context sentence, which is essential to many downstream tasks such as machine translation and information extraction. Recently, many approaches have achieved state-of-the-art performance on WSD by fine-tuning language models pretrained with massive text data on task-specific datasets (Blevins and Zettlemoyer, 2020; Yap et al., 2020).

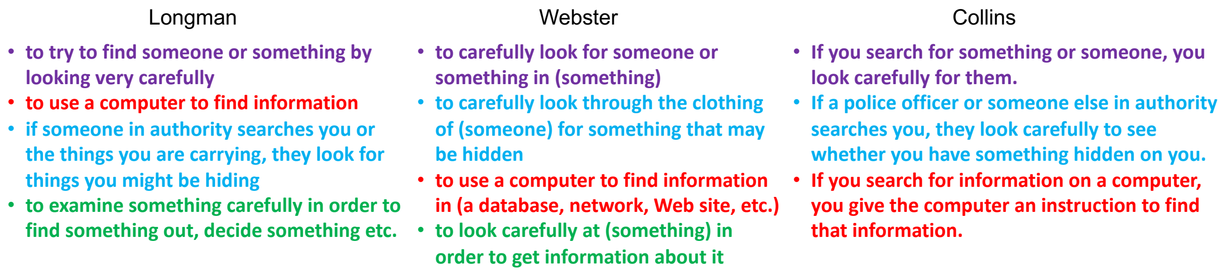
However, fine-tuning a WSD model using task-specific resources could limit its applicability and may cause two major problems. First, the performance of models decreases significantly when predicting on rare and zero-shot word senses (Kumar et al., 2019; Choubey and Huang, 2020; Blevins et al., 2021) because there are no sufficient supporting examples in training data. Second, the trained models are often inventory-dependent which can only select the best definition from one predefined word sense inventory (mainly WordNet) that human annotations are based upon.

In this paper, we overcome these limitations by leveraging abundant lexical knowledge from various word sense inventories. As we know, dictionaries that are compiled by experts contain rich sense knowledge of words. Moreover, a dictionary usually provides several example sentences for each word sense to illustrate its usage, which can be viewed as context sentences of that word sense. Since a word's sense (meaning) can be determined by its context, the word itself in a given context and the definition sentence corresponding to the correct sense are merely two surrogates of the same meaning (semantically equivalent). Furthermore, we observe that different dictionaries normally summarize meanings of a word to a close number of word senses, where definition sentences (glosses) from different dictionaries are different expressions of the same bunch of meanings. For example, Figure 1 lists glosses retrieved from three dictionaries for verb word *search*. We can see that glosses with the same color have the same meaning and can be aligned across different dictionaries.

Based on this observation, we propose a gloss alignment algorithm to leverage abundant lexical knowledge from various word sense inventories. We convert the problem of aligning two groups of glosses according to meanings to an optimization

[1] Models and code are available at https://github.com/wenlinyao/EMNLP21-ConnectTheDots. We will also release the checkpoint of the pretrained model for reproducibility.

Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 7741–7751
November 7–11, 2021. ©2021 Association for Computational Linguistics



Longman

- to try to find someone or something by looking very carefully
- to use a computer to find information
- if someone in authority searches you or the things you are carrying, they look for things you might be hiding
- to examine something carefully in order to find something out, decide something etc.

Webster

- to carefully look for someone or something in (something)
- to carefully look through the clothing of (someone) for something that may be hidden
- to use a computer to find information in (a database, network, Web site, etc.)
- to look carefully at (something) in order to get information about it

Collins

- If you search for something or someone, you look carefully for them.
- If a police officer or someone else in authority searches you, they look carefully to see whether you have something hidden on you.
- If you search for information on a computer, you give the computer an instruction to find that information.

Figure 1: Definition sentences of word *search* retrieved from three dictionaries: Longman Dictionary of Contemporary English, Merriam-Webster's Advanced Learner's Dictionary, and Collins COBUILD Advanced Dictionary.

problem – Maximum Weighted Graph Matching – to find the best matching that maximizes the overall textual similarity. In this way, we can gather general semantic equivalence knowledge from various dictionaries as a whole for all word senses, especially for rare senses that are less frequently seen in human-annotated data.

To make use of the derived semantic equivalence knowledge, we adopt a transfer learning approach that first pretrains a general semantic equivalence recognizer by contrasting the word representations in example sentences with the sentence representations of positive glosses or negative glosses. The general model can be directly applied to downstream WSD tasks or further fine-tuned on the task-specific dataset to get an expert model. We test our two-stage transfer learning scheme on two WSD benchmark tasks, i.e., the standard task (Raganato et al., 2017b) that focuses on all-words WSD and FEWS (Blevins et al., 2021) task that emphasizes low-shot (including few-shot and zero-shot) WSD. Experimental results show that the general model (without fine-tuning) surpasses the supervised baseline by 13.1% on zero-shot word senses. After further fine-tuning with build-in training data, the expert model outperforms the previous state-of-the-art model by 1.2% on all-words WSD tasks and 4.3% on low-shot WSD tasks. Adding semantic equivalence knowledge to the Word-in-Context (WiC) task (Pilehvar and Camacho-Collados, 2019) also improves the accuracy of RoBERTa$_{\text{Large}}$ (Liu et al., 2019) by 6%, which even outperforms the 9X larger T5 model (Raffel et al., 2020).

Overall, the major contributions of our work are two-fold. 1) We propose a gloss alignment algorithm that can integrate lexical knowledge from different word sense inventories to train a general semantic equivalence recognizer. 2) Without using task-specific training data, the general model not only performs well on overall word senses but demonstrates strong applicability to low-shot senses. The general model can turn into an expert model to achieve new state-of-the-art performance after further fine-tuning.

2 Related Work

Supervised WSD Approaches. Most existing WSD models are learned in a supervised manner and depend on human-annotated data. For example, Raganato et al. (2017a) regarded WSD as a sequence labeling task and trained a BiLSTM model with self-attention using multiple auxiliary losses. Luo et al. (2018a) introduced a hierarchical co-attention mechanism to generate gloss and context representations that can attend to each other. More recently, several BERT-based models have achieved new state-of-the-art performance on WSD by fine-tuning a pretrained language model. GlossBERT (Huang et al., 2019) appends each gloss to a given context sentence to create pseudo sentences and predicts them as either positive or negative depending on whether the sense corresponds to the correct sense or not. Bi-Encoder Model (BEM) (Blevins and Zettlemoyer, 2020) represents the target words and senses in the same embedding space using a context encoder and a gloss encoder but optimizes on each word individually. Yap et al. (2020) formulated WSD as a relevance ranking task and fine-tuned BERT to select the most probable sense definition from candidate senses. The neural architecture of our semantic equivalence recognizer realizes the benefits of GlossBERT and BEM.

Knowledge-Based WSD Approaches. Closely related to our work, many knowledge-based approaches rely on Lexical Knowledge Bases (LKB), such as Wikipedia and WordNet, to enhance representations of word senses. BabelNet (Navigli and Ponzetto, 2010) creates a resource by automatically mapping encyclopedic knowledge (Wikipedia) to lexicographic knowledge (WordNet) with the aid

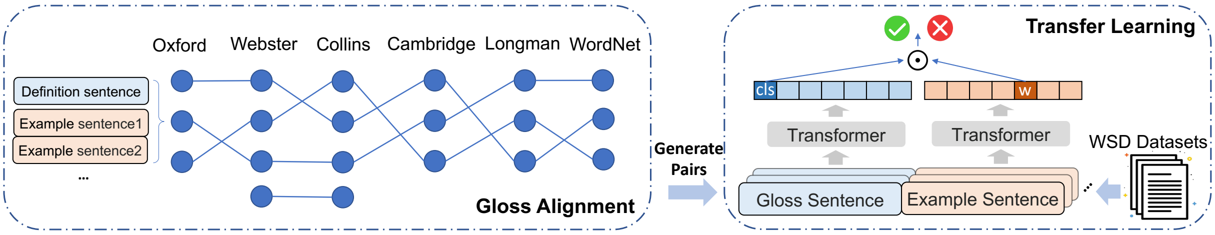


Figure 2: Overview of our approach. The left part illustrates the gloss alignment algorithm where each blue circle is a gloss containing one definition sentence and several example sentences. The right part is our model architecture to predict the semantic equivalence of a word in context and a gloss by comparing their representations obtained from a shared transformer encoder. Task-specific WSD datasets can be further used to fine-tune our model.

of Machine Translation. Lesk (Basile et al., 2014) relies on a word-level similarity function to measure the semantic overlap between the context of a word and each sense definition. SENSEMBERT (Scarlini et al., 2020a) produces high-quality latent semantic representations of word meanings by incorporating knowledge contained in BabelNet into language models. Other approaches try to learn better gloss embeddings by considering the WordNet graph structure (e.g., hypernyms, hyponyms, synonyms, etc.) (Luo et al., 2018b; Loureiro and Jorge, 2019; Kumar et al., 2019; Bevilacqua and Navigli, 2020). For example, Kumar et al. (2019) proposed EWISE to improve model's performance on rare or unseen senses by learning knowledge graph embeddings from WordNet. Building upon EWISE, Bevilacqua and Navigli (2020) developed a hybrid approach that incorporates more lexical knowledge (e.g., hypernymy, meronymy, similarity in WordNet) into the model through synset graph embeddings.

3 Overview of Our Approach

Figure 2 shows the overview of our approach. We first collect all word glosses and corresponding example sentences from six word sense inventories. We next apply the gloss alignment algorithm to find the best matching between two groups of glosses retrieved from two different inventories for every common keyword. By contrasting example sentences with the correct glosses and incorrect glosses within each inventory or across different inventories, we automatically gather rich supervision for pretraining a universal binary classifier that can determine whether the keyword in the context sentence (example sentence) and a gloss are semantically equivalent or not. The pretrained general model can be directly used in downstream WSD tasks or further fine-tuned to get an expert model.

Inventory	Words	Glosses	ES	Gls/W	ES/W
Oxford	52.5K	86.2K	96.8K	1.6	1.8
Webster	39.8K	72.5K	100.6K	1.8	2.5
Collins	34.4K	61.4K	89.5K	1.8	2.6
Cambridge	36.6K	67.0K	64.9K	1.8	1.8
Longman	36.9K	63.8K	70.2K	1.7	1.9
WordNet	147.3K	206.9K	47.4K	1.4	0.3

Table 1: Statistics of six word sense inventories used (phrases are included in word counting). ES: Example Sentences; Gls/W: average glosses per word; ES/W: average example sentences per word.

4 Aligning Glosses across Word Sense Inventories

4.1 Data Collection

We collected word sense inventory data by querying WordNet 3.0 (Miller, 1995) and the electronic edition of five professional dictionaries for advanced English learners: Oxford Advanced Learner's Dictionary (Turnbull, 2010), Merriam-Webster's Advanced Learner's Dictionary (Perrault, 2008), Collins COBUILD Advanced Dictionary (Sinclair, 2008), Cambridge Advanced Learner's Dictionary (Walter, 2008), and Longman Dictionary of Contemporary English (Summers, 2003). Advanced learners' dictionaries have a good characteristic that they usually provide abundant example sentences to illustrate the usage of different word senses in context, making it possible to generate strong supervision for training a classifier. Table 1 shows statistics of six word sense inventories used. In total, we collected 557.8K glosses and 469.4K example sentences.

4.2 Gloss Alignment as a Maximum-weight Matching Problem

Each word sense inventory is a lexical knowledge bank that provides example sentences for illustrating word senses, including senses less frequently seen in the real world. Moreover, we observe that different inventories usually provide parallel explanations of meanings for a given word (Figure 1). Thus, if we can align explanations (glosses) from different inventories according to meanings, we can significantly expand lexical knowledge acquired, especially for rare word senses. Essentially, finding the best alignment between two groups of glosses can be converted to Maximum-weight Bipartite Matching Problem (Cormen et al., 2009; Duan and Pettie, 2014) that aims to find a matching in a weighted bipartite graph that maximizes the sum of weights of the edges.

4.3 Problem Formulation

Given a keyword, suppose we retrieved two word sense sets S_1 and S_2 from two inventories, where each set consists of a list of definition sentences (glosses). Given a reward function r: $S_1 \times S_2 \to \mathbb{R}$, we want to find a matching[2] $f : S_1 \to S_2$ such that the total rewards $\sum_{a \in S_1, f(a) \in S_2} r(a, f(a))$ is maximized. By finding the matching f, we will know the best alignment between two word sense sets S_1 and S_2. In this paper, we use the sentence-level textual similarity as the reward function to find the best word sense alignment. To measure the textual similarity between two definition sentences, we apply a pretrained model SBERT (Reimers and Gurevych, 2019) that has achieved state-of-the-art performance on many Semantic Textual Similarity (STS) tasks and Paraphrase Detection tasks. Specifically, we apply SBERT to S_1 and S_2 to get sentence embeddings and then calculate cosine similarity as the reward function.

4.4 Solving Bipartite Graph Matching by Linear Programming

The Maximum-weight Graph Matching problem can be solved by Linear Programming (Matousek and Gärtner, 2007; Cormen et al., 2009). For simplicity, let weight w_{ij} denotes the textural similarity score between the i$^{\text{th}}$ definition sentence in S_1 and the j$^{\text{th}}$ definition sentence in S_2. We next introduce another variable x_{ij} for each edge (i, j). $x_{ij} = 1$ if the edge between i and j is contained in the matching and $x_{ij} = 0$ otherwise. The total weight of the matching is $\sum_{(i,j) \in S_1 \times S_2} w_{ij} x_{ij}$. To reflect every vertex is in exactly one edge in the matching, we add constraints $\sum_{j \in S_2} x_{ij} = 1$ for $i \in S_1$, and $\sum_{i \in S_1} x_{ij} = 1$ for $j \in S_2$, to guarantee that the variable x represents a perfect matching. Our goal is to find a maximum-weight perfect matching such that above constraints are satisfied. To sum up, aligning glosses between two word sense inventories is equivalent to solving the following linear integer programming problem:

$$\begin{aligned} \max_{\{x_{ij}\}} \quad & \sum_{(i,j) \in S_1 \times S_2} w_{ij} x_{ij} \\ \text{s.t.} \quad & \sum_{j \in S_2} x_{ij} = 1, i \in S_1 \\ & \sum_{i \in S_1} x_{ij} = 1, j \in S_2 \\ & x_{ij} \in \{0, 1\}, i \in S_1, j \in S_2 \end{aligned}$$

In our implementation, we consider all possible inventory combinations (select two from six) and apply the gloss alignment solver[3] to all common words shared by two inventories. For each word, the gloss alignment solver is only applied to glosses under the same POS category. Overall, we obtain 704K gloss alignment links.

4.5 Positive and Negative Training Instances

For a given word, the gloss alignment algorithm provides us the linking from word sense set S_1 in one inventory to S_2 in another inventory. Two glosses (e.g., $g \in S_1$ and $g' \in S_2$) have the same meaning if they are aligned by the algorithm or have a different meaning if they are not aligned. So we can pair the definition sentence of g (g') to each example sentence in g' (g) to generate gloss-context pairs for training the semantic equivalence recognizer. Pairs are labeled as positive if g and g' are aligned or negative otherwise[4]. In experiments, we only consider aligned gloss pairs with textual similarities above a threshold (see Section 6.1) to further improve the quality of supervision. In total, we generate 421K positive and 538K negative gloss-context pairs across different inventories.

Pairs are also generated by contrasting glosses within each inventory individually. In detail, for every word in an inventory, we pair the gloss sentence with its example sentences to get positive gloss-context pairs or pair the gloss sentence with example sentences from another gloss within the

[2] Note that unbalanced matching (i.e., S_1 and S_2 are different in size) can be reduced to balanced matching by adding new vertices to the smaller part and assigning weight 0 to edges pointing to them.

[3] Our implementation is based on Scipy library (`https://www.scipy.org/`).

[4] If S_1 and S_2 have a different number of glosses for a given word, we ignore the extra glosses that are not aligned.

inventory to get negative gloss-context pairs[5]. We generate 1.3M positive and 418K negative gloss-context pairs in this way. Similarly, we also generate context-context pairs by contrasting example sentences in two glosses to reflect the task setting of WiC (Section 6.3).

5 A Unified Neural Model for Recognizing Semantic Equivalence

5.1 Model Architecture

This section introduces our model architecture (the right part of Figure 2) for recognizing semantic equivalence. Inspired by Blevins and Zettlemoyer (2020), our model first uses an encoder to get the semantic representation of the target word (within its context sentence) or the gloss sentence. Next, by comparing two representations, our model predicts whether they are semantically equivalent or not.

Semantic Encoder. We apply a pretrained BERT model to get the contextual word representation of the target word (with its context) or the sentence representation of the gloss sentence. Specifically, given an input sentence S padded by the start symbol `[CLS]` and the end symbol `[SEP]`, we first obtain N contextualized embeddings $\{o_i\}_{i=1}^{N}$ for all tokens $\{t_i\}_{i=1}^{N}$ using BERT. We next select the contextualized embedding at the target word position[6] when S is a context sentence, or select the first output embedding o_0 (corresponding to the special token `[CLS]`) as the sentence representation when S is a gloss sentence.

Learning Objective. After deriving embeddings using BERT, both representations u and v, together with element-wise difference $|u - v|$ and element-wise multiplication $u \cdot v$ are concatenated and multiplied with the trained weight $W_t \in \mathbb{R}^{4n \times 2}$ with a softmax prediction layer for binary classification (semantically equivalent or not):

$$p = \text{softmax}(W_t[u, v, |u - v|, u \cdot v])$$

where n is the dimension of the embeddings. Our experiments consider two model sizes: **SemEq-Base** that is initialized with the pretrained BERT$_{\text{Base}}$ (Devlin et al., 2019) model with 12 transformer block layers, 768 hidden size, 12 self-attention heads and **SemEq-Large** that is initialized with the pretrained RoBERTa$_{\text{Large}}$ (Liu et al., 2019) model with 24 transformer block layers, 1024 hidden size, 16 self-attention heads[7]. We train our model using binary cross-entropy loss and AdamW (Loshchilov and Hutter, 2018) optimizer with initial learning rate {1e-5, 5e-6, 2e-6}, 0.2 dropout, batch size 64 and 10 training epochs.

	Noun	Verb	Adj	Adv	ALL
Percentage	55.6%	20.6%	20.2%	2.5%	100%
Accuracy	0.90	0.81	0.88	0.85	0.87

Table 2: Accuracy of the Gloss Alignment Algorithm.

6 Evaluation

6.1 Accuracy of the Gloss Alignment Algorithm

To evaluate the accuracy of the gloss alignment algorithm, we randomly sample 1,000 gloss pairs from 704K alignments and ask two human annotators to judge whether two gloss sentences refer to the same meaning or not. Two annotators labeled 200 gloss pairs in common and agreed on 94% (188) of them, achieving the kappa inter-agreement score of 0.74. One gloss pair is regarded as correct when both annotators label it as correct, and the remaining 800 gloss pairs were evenly allocated to two annotators to label. Table 2 shows the accuracy of the gloss alignment algorithm on each POS type based on human annotations. The accuracy on Noun, Verb, Adjective and Adverb words is 0.90, 0.81, 0.88 and 0.85, respectively, with an overall accuracy of 0.87. In experiments, we apply a threshold of 0.6 to alignment results and only consider aligned gloss pairs with textual similarities above it, which can further improve gloss alignment accuracy to 0.98 based on human annotations. In this way, we can significantly improve the quality of training data that are generated from the automatically aligned dictionaries.

6.2 Experiments on WSD

We evaluate our model on two WSD datasets, i.e., WSD tasks standardized by Raganato et al. (2017b) that focuses on all-words WSD evaluation and FEWS dataset proposed by Blevins et al. (2021) that emphasizes low-shot WSD evaluation. Since both datasets are annotated using word senses in WordNet 3.0 (Miller, 1995), we pair the context sentence with the annotated gloss in WordNet 3.0

[5] We only contrast to glosses having the same POS tag to get negative instances.

[6] If the target word is a phrase or the target word is tokenized into multiple subword pieces by the tokenizer, we average all subword embeddings to get its representation.

[7] Our implementation was based on https://github.com/huggingface/transformers.

		Model difference				Dev	Test				Concatenation of all Datasets				
	Models	TS	IK	GS	MS	SE07	SE2	SE3	SE13	SE15	Noun	Verb	Adj	Adv	ALL
1	Most Frequent Sense	✓	-	-	-	54.5	65.6	66.0	63.8	67.1	67.7	49.8	73.1	80.5	65.5
2	Lesk$_{\text{emb}}$ (2014)	✓	-	✓	-	56.7	63.0	63.7	66.2	64.6	70.0	51.1	51.7	80.6	64.2
3	BiLSTM (2017a)	✓	-	-	-	-	71.1	68.4	64.8	68.3	69.5	55.9	76.2	82.4	68.4
4	HCAN (2018a)	✓	-	-	-	-	72.8	70.3	68.5	72.8	72.7	58.2	77.4	84.1	71.1
5	EWISE (2019)	✓	-	✓	-	67.3	73.8	71.1	69.4	74.5	74.0	60.2	78.0	82.1	71.8
6	LMMS$_{\text{BERT}}$ (2019)	✓	-	✓	L	68.1	76.3	75.6	75.1	77.0	-	-	-	-	75.4
7	GlossBERT (2019)	✓	-	-	B	72.5	77.7	75.2	76.1	80.4	79.3	66.9	78.2	86.4	77.0
8	BEM (2020)	✓	-	-	B	74.5	79.4	77.4	79.7	81.7	81.4	68.5	83.0	87.9	79.0
9	AdaptBERT$_{\text{Large}}$ (2020)	✓	S	-	L	72.7	79.8	77.8	79.7	**84.4**	82.6	68.5	82.1	86.4	79.5
10	EWISER (2020)	✓	S	✓	L	**75.2**	80.8	79.0	80.7	81.8	82.9	69.4	**83.6**	87.3	80.1
11	SemEq-Base	✓	-	-	B	72.7	79.0	77.2	78.0	80.8	81.0	67.1	81.7	86.7	78.2
Ours: Data Augmentation															
12	SemEq-Base	✓	M	-	B	73.2	81.2	77.7	79.1	81.5	81.9	68.9	83.2	87.6	79.4
Ours: Transfer Learning															
13	SemEq-Base-General	-	M	-	B	65.7	75.3	70.9	78.0	79.8	78.2	61.3	81.2	80.3	74.8
14	SemEq-Base-Expert	✓	M	-	B	74.1	81.0	78.5	79.9	82.6	82.5	69.9	82.5	**88.4**	79.9
15	SemEq-Large-General	-	M	-	L	65.1	76.1	74.3	78.0	83.0	79.1	64.7	82.3	81.8	76.4
16	SemEq-Large-Expert	✓	M	-	L	74.9	**81.8**	**79.6**	**81.2**	81.8	**83.2**	**71.1**	83.2	87.9	**80.7**

Table 3: F1-score (%) on All-Words WSD benchmark datasets. We distinguish models based on 1) using the Training Set (TS) SemCor or not, 2) using single (S) Inventory Knowledge (IK) (i.e., WordNet) or our multi-source (M) inventory knowledge, 3) using WordNet synset Graph Structures (GS) or not, and 4) transformer Model Size (MS) of Base (B) or Large (L). Baseline systems are: Lesk$_{\text{emb}}$ (Basile et al., 2014), Babelfy (Moro and Navigli, 2015), BiLSTM (Raganato et al., 2017a), HCAN (Luo et al., 2018a), EWISE (Kumar et al., 2019), LMMS$_{\text{BERT}}$ (Loureiro and Jorge, 2019), GlossBERT (Huang et al., 2019), BEM (Blevins and Zettlemoyer, 2020), AdaptBERT$_{\text{Large}}$ (Yap et al., 2020), and EWISER (Bevilacqua and Navigli, 2020).

to generate positive gloss-context instances or other glosses of the word to get negative gloss-context instances for training. In validation or test, we apply the trained classifier to examine all possible glosses of the target word in WordNet 3.0 and select the gloss with the highest probability score as the prediction. To incorporate rich lexical knowledge harvested from word sense inventories into model training, we consider two strategies:

Data Augmentation. We directly augment the build-in training set from each WSD dataset with gloss-context pairs generated from our aligned word sense inventories and then train the semantic equivalence recognizer (SemEq) to do WSD.

Transfer Learning. We first train our semantic equivalence recognizer ONLY using gloss-context pairs generated from our aligned word sense inventories. The trained classifier is a general model (**SemEq-General**) capable of deciding whether a gloss sentence and the target word in a context sentence are semantically equivalent independent from any specific word sense inventories. Next, to evaluate on a specific WSD dataset, we further fine-tune the general model on the build-in training set to get an expert model (**SemEq-Expert**). The expert model can adapt to the new domain to achieve better performance.

6.2.1 All-Words WSD Tasks

We evaluate our model on the all-words WSD framework established by Raganato et al. (2017b). The testing dataset contains 5 benchmark datasets from previous Senseval and SemEval competitions, including Senseval-2 (SE2) (Edmonds and Cotton, 2001), Senseval-3 (SE3) (Mihalcea et al., 2004), SemEval-07 (SE07) (Pradhan et al., 2007), SemEval-13 (SE13) (Navigli et al., 2013), and SemEval-15 (SE15) (Moro and Navigli, 2015). Following Raganato et al. (2017b) and other previous work, we use SemCor (Miller et al., 1993) that contains 226,036 annotated instances as the build-in training set and choose SemEval-07 as the development set for hyper-parameter tuning. Since all datasets are mapped to word senses in WordNet 3.0 (Miller, 1995), we retrieve all definition sentences of the target word from WordNet 3.0 to form gloss-context pairs for both training and testing.

Table 3 shows experimental results on all-words WSD datasets (Raganato et al., 2017b). We also report models' performance on each POS category. The first section includes results of the most frequent sense baseline and previous WSD models.

The second section presents results of our model that adopt **data augmentation** strategy to incorporate multi-source inventory knowledge. SemEq-

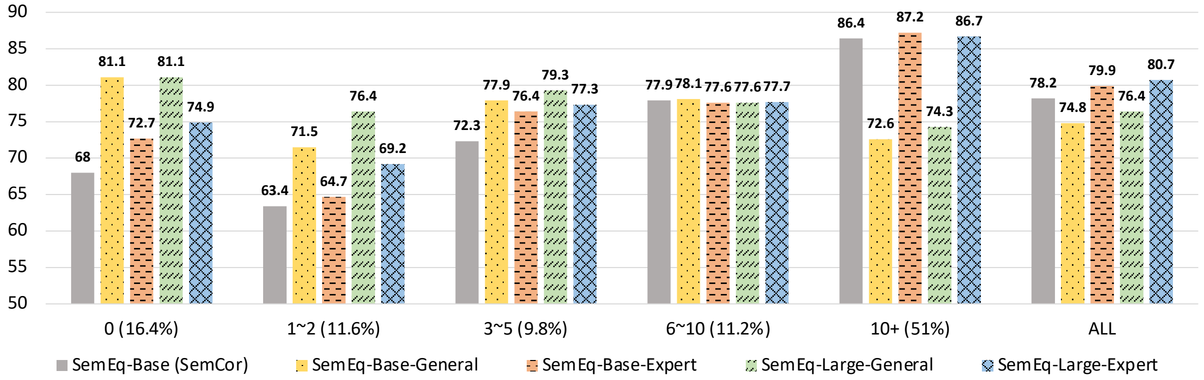


Figure 3: Evaluation (F1-score %) on the aggregated ALL set of All-Words WSD when we separate word senses based on their training instance numbers in SemCor.

Base (line 11) is our model's performance when fine-tuning $\text{BERT}_{\text{Base}}$ sentence encoder only on the build-in SemCor training set. Compared to line 11, when augmenting SemCor with our multi-source inventory knowledge, the same model (line 12) improves the F1 on the aggregated ALL set by 1.2%.

The third section of Table 3 reports the results of applying **transfer learning** strategy to exploiting our multi-source inventory knowledge. By only training on our multi-source inventory knowledge (without using SemCor), our model SemEq-Base-General (line 13) already achieves comparable performance with $\text{LMMS}_{\text{BERT}}$ (line 6, which is based on $\text{BERT}_{\text{Large}}$). After further fine-tuning on the training set - Semcor, SemEq-Base-Expert (line 14) improves the performance on ALL to 79.9%, which is slightly better than using the data augmentation strategy. Moreover, increasing BERT model parameters (line 16) further boosts the WSD performance on ALL to 80.7%[8].

Overall, our SemEq-Large-Expert model (line 16) consistently outperforms AdaptBERT (Yap et al., 2020) (line 9), the previous best model without using WordNet synset graph information, on SE07, SE2, SE3 and SE13, attaining 1.2% higher F1 on ALL. The SemEq-Large-Expert model also better disambiguates all types of words including nouns, verbs, adjectives, and adverbs than AdaptBERT. It clearly demonstrates the benefits of leveraging multiple word sense inventories via automatic alignment and transfer learning. Our final model is 0.6% higher even compared with EWISER (Bevilacqua and Navigli, 2020) that uses the *extra* WordNet graph knowledge. We can see that by pretraining on lexical knowledge derived from aligned inventories, our model generalizes more easily and better captures semantic equivalence between the target word and a gloss sentence for identifying the correct word meaning.

In order to understand our model's behavior of transferring semantic equivalence knowledge from our word sense inventories to a specific WSD task, we partition word senses in the test set into groups according to their numbers of training instances found in the training set SemCor. As shown in Figure 3, by pretraining on our semantic equivalence knowledge and then fine-tuning on SemCor, SemEq-Base-Expert beats SemEq-Base (SemCor) that is only trained on SemCor across all annotation-rich and annotation-lacking word senses. Interestingly, without fine-tuning on SemCor, the general model (SemEq-Base-General) works surprisingly well on low-shot senses, which is 13.1%, 8.1% and 5.6% higher than SemEq-Base (SemCor) on 0 shot, 1-2 shot, 3-5 shot senses, respectively. After fine-tuning on SemCor, the expert models fit to the distribution of senses in the real world and achieve better overall performance.

6.2.2 Few-Shot and Zero-Shot WSD Tasks

By pretraining on massive semantic equivalence knowledge generated from aligned word sense inventories, we expect our model performs better on annotation-lacking senses. We next evaluate our model on the FEWS dataset (Blevins et al., 2021), a new WSD dataset that focuses on low-shot WSD evaluation. FEWS is a comprehensive evaluation dataset constructed from Wiktionary and covers 35K polysemous words and 71K senses. Overall, the build-in training set of FEWS consists 87K sentence instances. The test (development) set consists of two evaluation subsets, i.e., a few-shot evalua-

[8] We also tried $\text{BERT}_{\text{Large}}$ which is slightly worse than $\text{RoBERTa}_{\text{Large}}$.

			Dev			Test		
	Models	TS	Full Set	Few-shot	Zero-shot	Full Set	Few-shot	Zero-shot
1	Most Frequent Sense	✓	26.4	52.8	0.0	25.7	51.5	0.0
2	Lesk$_{\text{emb}}$ (Basile et al., 2014)	✓	42.5	44.9	40.1	41.5	44.1	39.0
3	BEM(Blevins and Zettlemoyer, 2020)	✓	73.8	79.3	68.3	72.8	79.1	66.5
4	BEM$_{\text{SemCor}}$ (Blevins et al., 2021)	✓	74.4	79.7	69.0	73.0	78.9	67.1
5	SemEq-Base	✓	73.5	78.7	68.3	72.4	78.5	66.3
Ours: Data Augmentation								
6	SemEq-Base (+ WSI)	✓	74.2	78.4	69.9	73.7	78.6	68.7
Ours: Transfer Learning								
7	SemEq-Base-General	-	68.2	68.6	67.8	67.0	67.7	66.3
8	SemEq-Base-Expert	✓	76.0	80.4	71.5	75.2	80.1	70.2
9	SemEq-Large-General	-	70.7	70.9	70.5	69.8	71.2	68.4
10	SemEq-Large-Expert	✓	**77.8**	**81.8**	**73.7**	**77.3**	**82.3**	**72.2**

Table 4: F1-score (%) on the FEWS Low-Shot WSD benchmark dataset. WSI refers to knowledge extracted from aligned Word Sense Inventories. TS stands for the Training Set of FEWS.

Model	Acc.	Parameters
BERT$_{\text{Large}}$ (Devlin et al., 2019)	69.6	340M
RoBERTa$_{\text{Large}}$ (Liu et al., 2019)	69.9	355M
KnowBERT$_{\text{W+W}}$ (Peters et al., 2019)	70.9	523M
SenseBERT$_{\text{Large}}$ (Levine et al., 2020)	72.1	380M
T5-Large (Raffel et al., 2020)	69.3	770M
T5-3B (Raffel et al., 2020)	72.1	3000M
BERT$_{\text{ARES}}$ (Scarlini et al., 2020b)	72.2	342M
SemEq-Large (+WSI)	**75.9**	355M

Table 5: Accuracy (%) on the WiC benchmark dataset.

tion set and a zero-shot evaluation set; each subset contains 5K instances. Word senses that are used in zero-shot evaluation sets are verified to not occur in the training set, and word senses in few-shot evaluation sets will only occur 2 to 4 times in the training set.

Table 4 presents the results on FEWS. BEM$_{\text{SemCor}}$ (line 4) is a similar transfer learning model but fine-tuned on SemCor before training on FEWS while BEM (line 3) only trains on FEWS. The second section of Table 4 shows that augmenting the FEWS train set with our multi-source inventory knowledge (line 6) greatly improves zero-shot learning performance by 1.6% on the dev set and 2.4% on the test set (compared with line 5). Surprisingly, when we adopt the transfer learning strategy, the final SemEq-Large-Expert (line 10) model's performance on test sets increases to 82.3% on few-shot senses and 72.2% on zero-shot senses, which significantly outperforms all baseline models.

6.3 Experiments on Context-Sensitive Word Meanings

Word-in-Context (WiC) Task (Pilehvar and Camacho-Collados, 2019) from SuperGLUE benchmark (Wang et al., 2019) provides a high-quality dataset for the evaluation of context-sensitive word meanings. WiC removes predefined word senses and reduces meaning identification to a binary classification problem in which, given two sentences containing the same lemma word, a model is asked to predict whether the two target words have the same meaning. Considering WiC uses WordNet as one lexical resource in its data construction, we completely remove WordNet from our inventory knowledge to avoid data leaking. Specifically, we simply add context-context pairs[9] generated from the other five inventories to the training set of WiC to train a semantic equivalence recognizer. Table 5 shows results on the WiC task comparing to other models[10]. The results indicate that incorporating semantic equivalence knowledge from aligned inventories improves RoBERTa$_{\text{Large}}$'s performance by 6%, which also surpasses a large language model T5-3B (9X parameters) by 3.8%. It demonstrates the superiority of incorporating our high-quality multi-source lexical knowledge than blindly increasing the size of plain pretraining texts in language models.

7 Conclusion

Based on the observation that glosses of a word from different inventories usually are different expressions of a few meanings, we have proposed a gloss alignment algorithm that can unify different lexical resources as a whole to generate abundant semantic equivalence knowledge. The general model pretrained on derived equivalence knowledge can serve as a universal recognizer for word

[9] We generate 3.3M positive pairs and 1.7M negative pairs.

[10] We submit our model predictions to the competition page of WiC (`https://competitions.codalab.org/competitions`) to get the test results.

meanings in context or adapt to a specific WSD task by fine-tuning to achieve new state-of-the-art performance. Our results also point to an interesting future research direction: how to develop a robust fine-tuning approach that is able to retain the excellent performance of the general model on low-resource senses while still improving performance on high-resource senses.

Ethical Considerations

Copyrights of data used in this paper belong to their respective owners. The authors are permitted to use data under the permission of the non-commercial research purpose and following the principle of fair use. The authors will not reproduce, republish, distribute, transmit, or link data used on any other website without the express permission of respective owners. The authors bear the responsibility to comply with the rules of copyright holders.

References

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.

Michele Bevilacqua and Roberto Navigli. 2020. Breaking through the 80% glass ceiling: Raising the state of the art in word sense disambiguation by incorporating knowledge graph information. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2854–2864.

Terra Blevins, Mandar Joshi, and Luke Zettlemoyer. 2021. FEWS: Large-scale, low-shot word sense disambiguation with the dictionary. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 455–465, Online. Association for Computational Linguistics.

Terra Blevins and Luke Zettlemoyer. 2020. Moving down the long tail of word sense disambiguation with gloss informed bi-encoders. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1006–1017.

Prafulla Kumar Choubey and Ruihong Huang. 2020. One classifier for all ambiguous words: Overcoming data sparsity by utilizing sense correlations across words. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 5978–5985.

Thomas H Cormen, Charles E Leiserson, Ronald L Rivest, and Clifford Stein. 2009. *Introduction to algorithms*. MIT press.

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.

Ran Duan and Seth Pettie. 2014. Linear-time approximation for maximum weight matching. *Journal of the ACM (JACM)*, 61(1):1–23.

Philip Edmonds and Scott Cotton. 2001. Senseval-2: overview. In *Proceedings of SENSEVAL-2 Second International Workshop on Evaluating Word Sense Disambiguation Systems*, pages 1–5.

Luyao Huang, Chi Sun, Xipeng Qiu, and Xuan-Jing Huang. 2019. Glossbert: Bert for word sense disambiguation with gloss knowledge. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3500–3505.

Sawan Kumar, Sharmistha Jat, Karan Saxena, and Partha Talukdar. 2019. Zero-shot word sense disambiguation using sense definition embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5670–5681.

Yoav Levine, Barak Lenz, Or Dagan, Ori Ram, Dan Padnos, Or Sharir, Shai Shalev-Shwartz, Amnon Shashua, and Yoav Shoham. 2020. Sensebert: Driving some sense into bert. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4656–4667.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Ilya Loshchilov and Frank Hutter. 2018. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Daniel Loureiro and Alipio Jorge. 2019. Language modelling makes sense: Propagating representations through wordnet for full-coverage word sense disambiguation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5682–5691.

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.

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.

Jiri Matousek and Bernd Gärtner. 2007. *Understanding and using linear programming*. Springer Science & Business Media.

Rada Mihalcea, Timothy Chklovski, and Adam Kilgarriff. 2004. The senseval-3 english lexical sample task. In *Proceedings of SENSEVAL-3, the third international workshop on the evaluation of systems for the semantic analysis of text*, pages 25–28.

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.

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.

Roberto Navigli and Simone Paolo Ponzetto. 2010. Babelnet: Building a very large multilingual semantic network. In *Proceedings of the 48th annual meeting of the association for computational linguistics*, pages 216–225.

Stephen J. (ed.) Perrault. 2008. *Merriam-Webster's Advanced Learner's English Dictionary*. Springfield, MA: Merriam-Webster.

Matthew E Peters, Mark Neumann, Robert Logan, Roy Schwartz, Vidur Joshi, Sameer Singh, and Noah A Smith. 2019. Knowledge enhanced contextual word representations. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 43–54.

Mohammad Taher Pilehvar and Jose Camacho-Collados. 2019. Wic: the word-in-context dataset for evaluating context-sensitive meaning representations. 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 1267–1273.

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.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21:1–67.

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

Alessandro Raganato, Jose Camacho-Collados, and Roberto Navigli. 2017b. 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.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3973–3983.

Bianca Scarlini, Tommaso Pasini, and Roberto Navigli. 2020a. Sensembert: Context-enhanced sense embeddings for multilingual word sense disambiguation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8758–8765.

Bianca Scarlini, Tommaso Pasini, and Roberto Navigli. 2020b. With more contexts comes better performance: Contextualized sense embeddings for all-round word sense disambiguation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3528–3539.

John (ed.) Sinclair. 2008. *Collins COBUILD Advanced Dictionary*. London: HarperCollins.

Della (ed.) Summers. 2003. *Longman Dictionary of Contemporary English*. Harlow: Pearson Education.

Joanna (ed.) Turnbull. 2010. *Oxford Advanced Learner's Dictionary*. Oxford: Oxford university press.

Elizabeth (ed.) Walter. 2008. *Cambridge Advanced Learner's Dictionary*. Cambridge: Cambridge university press.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. 2019. Superglue: A stickier benchmark for general-purpose language understanding systems. *Advances in Neural Information Processing Systems*, 32.

Boon Peng Yap, Andrew Koh, and Eng Siong Chng. 2020. Adapting bert for word sense disambiguation with gloss selection objective and example sentences. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 41–46.