ECAI 2020
G.D. Giacomo et al. (Eds.)

doi:10.3233/FAIA200359

Leveraging Human Prior Knowledge to Learn Sense Representations

Tong Zhang and **Wei Ye*** and **Xiangyu Xi** and **Long Zhang** and **Shikun Zhang** and **Wen Zhao** [1]

Abstract. Conventional distributed word representation learning, which learns a single vector for each word, is unable to represent different meanings of polysemous words. To address this issue, a number of approaches were proposed to model individual word senses in recent years. However, most of these sense representations are hard to be integrated into downstream tasks. In this paper, we propose a knowledge-based method to learn word sense representations that can offer effective support in downstream tasks. More specifically, we propose to capture the semantic information of prior human knowledge from sememes, the minimum semantic units of meaning, to build global sense context vectors and perform a reliable soft word sense disambiguation for polysemous words. We extend the framework of Skip-gram model with a contextual attention mechanism to learn an individual embedding for each sense. The intrinsic experimental results show that our proposed method can capture the distinct and exact meanings of senses and outperform previous work on the classic word similarity task. The extrinsic experiment and further analysis show that our sense embeddings can be utilized to effectively improve performance and mitigate the impact of polysemy in multiple real-word downstream tasks.

1 Introduction

Distributed word representation has been widely used in natural language processing (NLP) due to its ability to capture semantic information of words, mostly as a fundamental step in neural approaches. The main idea is to represent each word with a dense vector in a continuous low-dimensional semantic space [31] where words with similar meanings are close to each other. The most popular approaches of word representation learning are based on context tokens prediction [19] and co-occurrence matrix factorization [25].

However, many words are polysemous, which means they have multiple senses. Since conventional word representation learning methods represent each word with a single vector, these word embeddings suffer from the meaning conflation deficiency problem [32] and lack the ability to capture the exact semantic meanings of different senses [30]. Moreover, the meaning conflation would shorten the distance between words with different meanings in semantic space when they are similar to the different senses of another word [22], which impacts the effectiveness of semantic space [33].

To address this issue, some research on learning multiple representations for senses of a word was presented in recent years. One important branch is unsupervised sense representation, which induces distinctive senses of a word from the corpus without supervision [29, 13, 22, 17]. For these unsupervised approaches, a drawback is that the induced senses are hard to distinguish and cannot correspond with the meanings in the real world. Additionally, the induced senses have tight correlation with the training corpus, leading to an ordinary performance [18, 2] in downstream tasks where the distribution of corpus is different. Another branch is knowledge-based sense representation [4, 15, 3, 27], which constructs senses by exploiting knowledge resources and learn sense representations that explicitly correspond to a specific word sense defined in the knowledge resource.

Word sense disambiguation is a crucial step in sense representation learning. However, in most existing approaches, the word sense disambiguation step is simply based on the similarity between sense embeddings and context embeddings. The semantic information in the knowledge resource is underutilized as it is only used to initialize sense embeddings. In addition, most existing approaches perform a **hard disambiguation** step, in which the most appropriate sense is selected to represent a word. However, different senses of a word are not completely independent to each other and they may jointly contribute to the word meaning in certain context.

Niu et al. [24] first proposed a knowledge-based method to learn both Chinese word and sense representation, which utilized attention mechanism and the knowledge resource to perform **soft disambiguation**. Though their enhanced word embeddings achieved great performance in intrinsic evaluation, according to our verification in Section 6, their by-product sense embeddings were unable to capture the exact semantic information for senses, which means such sense embeddings could not eliminate the meaning conflation deficiency problem and support downstream tasks. The reason is probably that the word sense disambiguation step is unreliable as it utilized the hidden representations of sense and context to perform word sense disambiguation, which are even not in a same semantic space.

The recent contextualized representations such as ELMo [26], GPT [28] and BERT [6] provide each token a context-dependent representation, which have been proven to be effective for improving the performances in various NLP tasks. These representations could implicitly alleviate the impact of polysemous words to some extent, because different senses have different contexts in most cases. However, in contrast to sense embeddings, contextualized representations could not explicitly distinguish the polysemous word. Besides, due to the complex model with large number of parameters, in some case contextualized embeddings are problematic in practice. For example, They are inapplicable in some real-time systems or resource-restricted systems because of the low inference-time efficiency and large memory usage. The large-scale corpus to train such large models are also difficult to obtain in some areas. Thus, learning light-weighted sense representations that is beneficial to downstream tasks

[1] National Engineering Research Center for Software Engineering, Peking University, China, email: {zhangtong17, wye, xixy, zhanglong418, zhangsk, zhaowen}@pku.edu.cn
*Corresponding Author

is still a valuable work.

In this paper, we aim to leverage knowledge resources to disambiguate word senses and learn sense representations that are beneficial to downstream tasks. Following Niu et al. [24], we adopt HowNet [7] as the sense inventory, which is a Chinese lexical knowledge base widely used in such NLP tasks as word similarity calculation [5], word representation learning [24] and language modeling [10]. In HowNet, each word is annotated with one or multiple senses and the meaning of each sense is unveiled by one or multiple sememes, which are the minimum semantic units of meaning [1]. The detail of HowNet is in Section 3.1.

We propose a novel Sememe-based Contextual Attention (SCA) model to learn individual sense representations with HowNet. SCA model utilizes the semantic information of sememe knowledge to constitute the global context of each sense, which is used to model the global context distribution of a sense in the corpus. Once the global sense contexts are constituted, they are fixed and used to perform soft word sense disambiguation with a contextual attention mechanism. Our reliable word sense disambiguation step utilizes the context word distribution rather than the hidden representation of the context and the senses of target words, which are even not in a same semantic space and carry no semantic information at the start of training. The sense embeddings learned by SCA can be easily integrated to downstream NLP tasks with an additional soft disambiguation step. We conduct both intrinsic evaluation and extrinsic evaluation on our model. The experimental results show that the sense embeddings by our model can: (1) effectively capture semantic information and build high-quality sense representations, (2) provide strong support in downstream tasks by addressing polysemy issue.

To summarize, we make the following contributions in this paper:

- We propose a novel approach to perform reliable soft sense disambiguation with contextual attention and learn knowledge-based sense embeddings, which achieve significant improvement in intrinsic evaluation.
- Our sense embedding can be easily integrated into downstream tasks because of our reliable sense disambiguation step with the contextual attention mechanism.
- We evaluate the effectiveness of the sense embeddings learned by our approach in token-level, sentence-level and document-level NLP tasks, and the results show that these sense embeddings are beneficial for downstream tasks by mitigating the impact of polysemy.

2 Related Work

2.1 Unsupervised Sense Representation Learning

Reisinger and Mooney [29] proposed a method to build multi-prototype representations for words by clustering the occurrences of each word, where senses are represented by the cluster centroids. Following that, Huang et al. [13] introduced a cluster-based neural language model to learn sense representations. In these methods, the number of senses is fixed for each word, which is unrealistic.

Neelakantan et al. [22] proposed to learn sense representation and disambiguation jointly with a extended Skip-gram model, with a varying number of senses per word. Lee and Chen [17] first proposed a modularized framework based on reinforcement learning to learn sense representations with a separated sense selection module. Li and Jurafsky [18] tested the performance of unsupervised sense representations on natural language understanding tasks and observed improvement in some of the tasks (part-of-speech tagging, semantic relation identification, semantic relatedness).

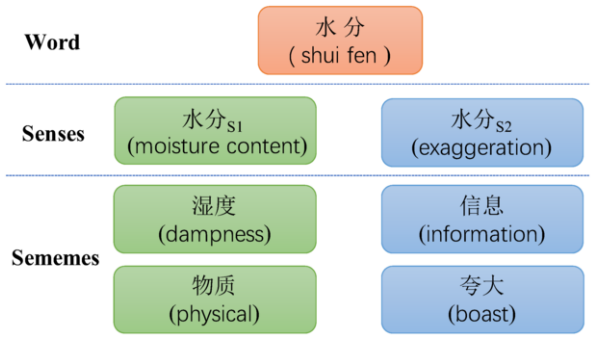

Figure 1. The definition of the two senses of "水分" in HowNet.

2.2 Knowledge-enhanced Representation Learning

A series of methods were proposed to utilize the glosses in WordNet [21] to initialize sense vectors with the average word embeddings of glosses [4], or representation of glosses generated by a convolutional neural network [3], and perform word sense disambiguation to learn sense specific representation. Yang and Mao [34] proposed a supervised fine tuning framework to learn multi-prototype embeddings from existing word embeddings and mini-contexts of word pairs. This post-process method provides a new solution, in which the sense embeddings are transformed from word embeddings with corrupted information [12].

In addition to these methods, SE-WRL [24] first proposed to integrate HowNet [7] to jointly learn representations of Chinese words, senses and sememes. They applied attention scheme to disambiguate senses by calculating the similarity between sense embeddings and context embeddings. The knowledge-enhanced word representations from SE-WRL achieved great performance in intrinsic evaluation. However,the sense embeddings from SE-WRL failed to model the exact meaning for each sense, hence this method cannot fully eliminate the meaning conflation deficiency problem.

3 Background

3.1 HowNet

HowNet [7] is one of the most widely used fully computational Chinese knowledge base, unveiling the meaning of concepts in lexicons with sememes. In HowNet, words, senses and sememes are organized into three top-down levels. Each word is annotated by one or multiple senses, and each sense is annotated by a set of sememes. A sememe is defined as the minimum semantic unit, summarized manually. Figure 1 shows the senses and sememes of the polysemous word "水分". HowNet defines two common senses of "水分", which are "*moisture content*" and "*exaggeration*". Each sense is interpreted by a set of sememes. For the first sense "*moisture content*", its semantic definition is given by the sememes "湿度" (dampness) and "物质" (physical), which can be glossed thus: the dampness in physical stuff. Besides, the other sense "*exaggeration*" is defined by the sememes "信息" (information) and "夸大" (boast).

HowNet interprets more than 110,000 senses of words with only 1983 sememes. Senses with similar meanings are annotated by sim-

ilar sets of sememes. In Chinese, there have been a number of researches on measuring semantic similarity between words or senses [11, 8, 5, 9].

3.2 Skip-gram Model

The Skip-gram model, proposed in Word2Vec [19], is one of the most widely used model in word representations learning because of its efficiency and performance of semantic modeling. This model is basically a log-linear classifier with two projection, which aims at predicting the surrounding context words of the target word. The probability of context word w_c, based on target word w is given by:

$$P(w_c|w) = \frac{\exp(v_{w_c}^\top \cdot v'_w)}{\sum_{w'_c \in W} \exp(v_{w'_c}^\top \cdot v'_w)} \quad (1)$$

where v_w presents the embedding vector for w from the input embedding matrix and v'_w is from the output embedding matrix. For each target word w, given a context words set C sampled from a dynamic sliding window K, the model aims to minimize the following loss function:

$$L = -\sum_{w_c \in C} \log(P(w_c|w))) \quad (2)$$

4 Methodology

In this section, we present the **S**ememe-based **C**ontextual **A**ttention (SCA) model for learning multiple embeddings for polysemous word, each of which corresponds to a specific word sense defined by sememes in HowNet. Our approach follows the conventional Skip-gram model [19], with the addition of utilizing sememes and contextual information to disambiguate and represent word senses.

In the following sections, we denote the total vocabulary as *W*. For each word *w* in *W*, S_w is the sense set and $s_i \in S_w$ is the i^{th} sense of word *w*. Each sense *s* is defined by a sememe set M_s, where m_s^i represents the i^{th} sememe of sense *s*.

4.1 SCA Model

The overall architecture of SCA is shown in Figure 2, which contains three stages as follows:

- **Global word context generation.** For each word defined in HowNet, we generate the global context derived from its occurrences in a large textual corpus.
- **Global sense context generation.** Given a target word, we first calculate the similarity matrix for its senses. Then we generate the global context for each sense, constituted by the global context of its similar words.
- **Soft word sense disambiguation and sense representation learning.** With the pre-generated global sense context matrix, we perform reliable soft word sense disambiguation on target word by calculating the contextual attention for its senses.

4.1.1 Global Word Context Generation

Considering that senses are not directly available in unlabeled corpus, the basic idea of SCA model is to disambiguate polysemous words with the global context of each sense, which constituted by the global context the of words having similar meaning to the target sense. To achieve that, we first generate the global word context matrix C^w from the corpus to represent the statistical context of each word. More specifically, C^w represents the matrix of global co-occurrence counts, where each row C_i^w is a bag-of-words and each element C_{ij}^w represents the counts w_j occurs within the context window K of w_i. Additionally, since frequent words are less representative to disambiguate word senses, we weight C_{ij}^w with a subsampling weight $sub(w_j)$ inherited from Mikolov et al. [20], computed as follows:

$$sub(w_j) = \min(\sqrt{\frac{\delta}{f(w_j)}}, 1) \quad (3)$$

where δ is the subsampling threshold and $f(w_j)$ is the frequency of w_j in corpus. $sub(w_j)$ is the probability for w_j to remain during subsampling in the Skip-gram model. Finally, we perform L2 normalization on each row of C^w:

$$C_i^w \leftarrow \frac{C_i^w}{||C_i^w||} \quad (4)$$

4.1.2 Global Sense Context Generation

In this step we generate the global context for each sense of the target word. We describe following two matrixes, which are the kernel parts in this step.

Sense-word similarity matrix. Assuming that words with similar meanings have similar distributions, we represent the context distributions of each sense with the context distributions of words which have similar meaning to the target sense. Thus we propose a simple algorithm to compute the similarity between senses and words. Formally, given a sense s and a word w, the similarity between s and w can be defined as:

$$\mathrm{Sim}(s, w) = \max_{s_i \in S_w} \{\frac{\mathrm{sim}(s, s_i)}{\sqrt{|S_w|}}\} \quad (5)$$

where $|S_w|$ denotes the number of senses of w and $\mathrm{sim}(s_1, s_2)$ denotes the similarity between s_1 and s_2, which is measured by Ochiai coefficient as follows:

$$\mathrm{sim}(s_1, s_2) = \frac{|M_{s_1} \bigcap M_{s_2}|}{\sqrt{|M_{s_1}| \times |M_{s_2}|}} \quad (6)$$

Given a target word w_t and its senses to be disambiguated, we generate the sense-word similarity matrix $Sim(w_t)$, where $Sim(w_t)_{ij}$ denotes the similarity between sense s_i and word w_j. Furthermore, for each sense, we retain its similarity to the top N similar words and zero its similarity to other words in order to accelerate learning and reduce the noise brought by the words with deviated meaning. N is set to be 5 straightforwardly according to our observation.

Global sense context matrix. After generation of sense-word similarity matrix, we can utilize $Sim(w)$ as the weight matrix to weight the context vectors in C^w and build the statistic global context vector for each sense. Formally, given a sense s_i of word w, the global context vector is calculated as follows:

$$C^s(s_i) = \sum_{j=1}^{|W|} Sim(w)_{ij} \cdot C_j^w \quad (7)$$

where $C^s(s_i)$ represents the global context vector of sense s_i and it is one row of the global sense context matrix $C^s(w)$ of the target word w.

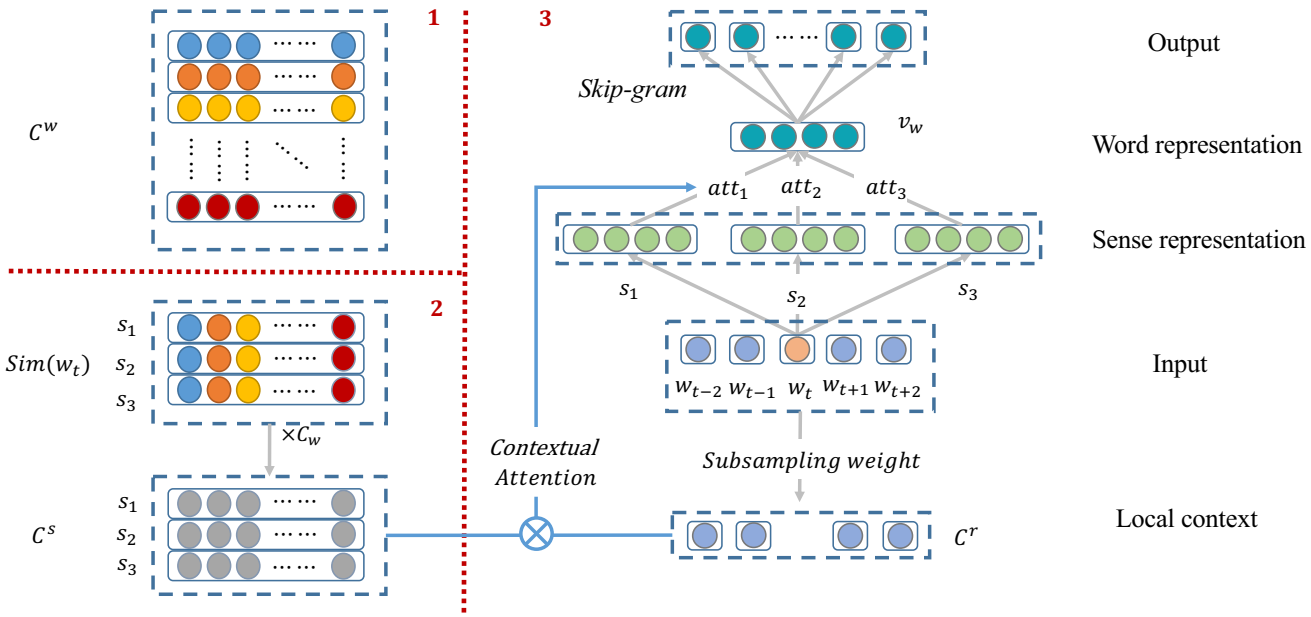

Figure 2. The architecture of SCA model. The red numbers represent the three stages in Section 4.1. w_t is the target word and s_i is the i^{th} sense of w_t. C^w represents the global word context matrix. Each row of C^w denotes the global context vector of a specific word, which is derived from a large textual corpus. $Sim(w_t)$ denotes the sense–word similarity matrix of target word w_t. C^s represents the global sense context matrix, where each row is the global context vector for one sense of w_t. C^r denotes the local context of w_t in a training instance.

4.1.3 *Sense Representation Learning with Soft Disambiguation*

With the global sense context matrix, we can perform word sense disambiguation with a contextual attention mechanism. For each sense $s_i \in S_w$, the contextual attention scheme calculates a attention score a_i, which indicates the normalized similarity between the local context vector C^r and the context vector of each sense $C^s(s_i)$. We calculate the attention score a_i of s_i as following:

$$e_i = C^s(s_i) \cdot C^{r\top} \tag{8}$$

$$a_i = \frac{\exp(\gamma e_i)}{\sum_{j=1}^{|S_w|} \exp(\gamma e_j)} \tag{9}$$

where γ is the scale coefficient. The algorithm can be considered as soft word sense disambiguation if γ is not particularly large.

After word sense disambiguation, we extend Skip-gram model with a contextual attention mechanism to learn separated representation for each sense, which is regarded as the minimum unit in the sentence. We utilize contextual attention to select the appropriate senses to make up the target word representation v_w, which is formalized as follows:

$$v_w = \sum_{i=1}^{|S_w|} a_i \cdot v_{s_i} \tag{10}$$

where v_{s_i} denotes the embedding vector of $s_i \in S_w$ and a_i is the attention score for s_i. Finally, v_w is used to predict the context words by Noise Contrastive Estimation introduced in [20], which can reduce the computational complexity significantly.

4.2 Usage of Sense Representations for Downstream Tasks

In this section, we describe how to integrate our sense embeddings in downstream tasks.

In many downstream NLP tasks, word representation is used as the foundational module to convey semantic information in words. Since word sense is not directly obtainable in downstream tasks, we perform a soft word sense disambiguation step to select appropriate word senses and represent words with sense embeddings weighted by contextual attention, same as the process in the previous section. Specifically, we use Equation 7 to build the sense context matrix $C^s(w)$ for word w with word context matrix C^w and calculate contextual attention according to Equation 8 and 9. The representation of w is built of the sense embeddings with contextual attention, as shown in Equation 10. Note that the representation of each word is not fixed when used in downstream tasks because of the soft word sense disambiguation with contextual attention, thus our sense embedding can bring additional contextual information that is supportive to downstream tasks.

5 Intrinsic Evaluation

In this section, we explore the intrinsic quality of our method with word similarity task and a qualitative investigation. Firstly we evaluate the sense embeddings from our SCA model on the word similarity task, showing our embedding improves the correlation with human judgments. Next, we perform a qualitative investigation on nearest neighbors and word sense disambiguation of our method.

5.1 Experiment Settings

For word context matrix generation and sense representation learning, we adopt SogouCS[2] as the corpus. SogouCS is a Chinese news dataset, which contains about 1.3 million news and 1.9 billion characters. In preprocessing, we remove non-Chinese characters and perform Chinese word segmentation with Stanford CoreNLP[3] toolkit. We use HowNet[4] as the sense inventory. In HowNet, 1,983 distinct sememes are defined manually to unveil the meaning of word senses. According to the sense inventory, 52.22% of words in preprocessed SogouCS are polysemous, showing the significance to learn sense representations respectively.

Three conventional methods, including GloVe, CBOW and Skip-gram model, are chosen as the baselines. Additionally, due to the Chinese corpus and knowlege base, we compare our SCA model with both word embeddings and sense embeddings from the best model SAT of SE-WRL [24]. To the best of our knowledge, SE-WRL is the exclusive method to learn Chinese word and sense representations with monolingual corpus.

For all experiments, we select the best hyper-parameters with same corpus and experimental settings. Specifically, we set the dimensions of all word and sense embeddings to be 256. For SCA model, the size of context window K is only fixed as 4 when calculating the contextual attention, whereas it is dynamic during training with a max size 4. The vocabulary size is set to be 160,000. We set the negative samples to be 64. The sub-sampling parameter δ is 10^{-4}. The learning rate lr is initialized to be 0.2, and will decay through the total training 5 iterations. Additionally, the scale coefficient γ is 27. During word context matrix generation we set N to be 5, and context window increases to 8 because sub-sampling approach is abandoned in this step.

Table 1. Spearman correlation $\rho \times 100$ on word similarity task. Hard represent the SCA model with contextual hard attention, which only select the most appropriate sense in word sense disambiguation. Soft represent the SCA model with contextual soft attention.

Model	Wordsim-240	Wordsim-297
Glove	57.73	55.16
CBOW	55.20	58.16
Skip-gram	55.69	58.91
SAT (word)	57.21	59.27
SAT (sense)	2.16	5.80
SCA (hard)	58.62	59.35
SCA (soft)	**60.14**	**61.33**

5.2 Word Similarity

We choose wordsim-240 and wordsim-297[5] to evaluate the performance of word similarity computation, which were provided by Chen et al. [3]. According to our statistics, about 30% words are polysemous in wordsim-240 and wordsim-297 by HowNet definition. For word embedding evaluation, we use the cosine similarity to sort all the word-pairs and compare the orderings of models against the one obtained by the human judgments with the Spearman correlation ρ. For sense embedding evaluation, we measure the similarity between words based on MaxSim metric [29], which is the most popular method to adapt word-based similarity benchmarks to sense:

$$\mathrm{MaxSim}(w, w') = \max_{s \in S_w, s' \in S_{w'}} \cos(v_s, v_{s'}) \tag{11}$$

The Spearman correlation results are shown in Table 5.1. We observe that representations leveraging sense information including SAT(word) and SCA outperform conventional word embeddings, which indicates that prior human knowledge of word senses is beneficial to model exact meanings of words or senses. The sense embeddings of SAT offer a poor performance, which means SAT can't capture the exact meaning of each sense. The reason may lie in that sense disambiguation with hidden representations is not completely reliable and it's weak to represent sense with the average of sememe representations. In contrast, our sense embeddings with soft attention achieves a significant improvement compared with the word embeddings of SAT, which indicates that our model can make better use of sense knowledge to disambiguate and model word senses. The performance decline of SCA sense embeddings with hard attention demonstrate the effectiveness of soft word sense disambiguation. The reason is that senses of a word are not always completely separated from each other, whereas they are related and constitute the meaning of the word jointly in some context.

5.3 Qualitative Investigation

We first perform a qualitative investigation on nearest neighbors using conventional Skip-gram model and our SCA model. As shown in Table 2, as opposed to conventional word embeddings, our sense embeddings can capture the exact meaning of each sense. For example, the word "摩擦" has two senses, including "*rub*" and "*conflict*". We find that "*abrasion*" and "*brawl*" are the top two nearest neighbors of "摩擦" according to word embeddings, which indicates that word embeddings may mix the meanings of different senses of a polysemous word. Besides, some word embeddings would be partial to one of the senses, which can be seen from the nearest neighbors of "苹果". In contrast, the nearest neighbors of senses computed by sense embeddings are more accurate and unambiguous.

In further investigation, our sense embeddings are proven to mitigate the meaning conflation deficiency problem effectively. For example, "*banana*" ranks 700th among the nearest neighbors of "*Google*" according to word embeddings, because both "*Google*" and "*banana*" are similar to the different sense of "苹果" (apple). In contrast, "*banana*" is far from "*Google*" with the ranking of 27,098th according to our sense embeddings.

In addition, Table 3 shows some examples of soft sense disambiguation in certain contexts. For each word, the first row shows the sememes of each sense and the senses of each word. The attention score of each sense in a particular context is given in the table. Obviously, our model can effectively choose the appropriate senses for word in context with soft attention scores.

6 Extrinsic Evaluation

We explore the extrinsic effectiveness of our sense embeddings across three downstream tasks in different levels: event detection (word level), relation classification (sentence level) and text classification (document level). Since SAT focus on intrinsic evaluations and we find no obvious difference between the performance of word embeddings of SAT and conventional word embeddings in downstream tasks, we choose Skip-gram as the baseline. For each task, we adopt the following embedding settings and experiment on them respectively without other modifications:

[2] http://www.sogou.com/labs/resource/cs.php
[3] https://nlp.stanford.edu/software/segmenter.shtml
[4] http://www.keenage.com
[5] https://github.com/Leonard-Xu/CWE/tree/master/data

Table 2. Nearest neighbors of word embeddings and sense embeddings.

Word/Sense (W/S)	Nearest Neighbors
W: **摩擦(rub/conflict)**	磨损/abrasion, 争吵/brawl, 口角/quarrel, 矛盾/contradiction, 纠纷/dispute, 撞击/knock
S_1: **磨损(rub)**	磨擦/friction, 磨损/abrasion, 色牢/color fast, 牢度/fastness, 耐/durability, 刮擦/scratch
S_2: **冲突(conflict)**	撕扯/rend, 厮打/tussle, 矛盾/contradiction, 不和/disharmony, 争执/dispute, 闹/fracas
W: **苹果(Apple brand/apple)**	微软/Microsoft, 三星/Samsung, 谷歌/Google, 黑莓/BlackBerry, 摩托罗拉/Motorola
S_1: **电脑(Apple brand)**	微软/Microsoft, 谷歌/Google, 三星/Samsung, 摩托罗拉/Motorola, 诺基亚/Nokia
S_2: **水果(apple)**	香蕉/banana, 果品/fruit, 猕猴桃/kiwifruit, 桃子/peach, 葡萄/grape, 果农/fruit farmer

Table 3. Examples of word sense disambiguation in context.

W**: 摩擦 (**S_1**:conflict,** S_2**:rub)** S_1: ("fight")	S_2: ("rub")	
双手**摩擦**会让手心变热 (**Rubbing** hands will warm your palms.)	*conflict*: 0.18	*rub*: 0.82
两个人在一起有**摩擦**是正常的 (It's normal that **conflicts** appear between people in love.)	*conflict*: 0.80	*rub*: 0.20
W**: 苹果 (**S_1**:Apple brand,** S_2**:apple)** S_1: ("computer","able","PatternValue", "bring","SpeBrand")	S_2: ("fruit")	
今年的富士**苹果**吃起来又甜又脆 (Fuji **apple** taste sweet and crisp this year.)	*Apple brand*: 0.08	*apple*: 0.92
新出品的**苹果**手机功能很全 (**Apple**'s new phone has complete functions.)	*Apple brand*: 0.87	*apple*: 0.13

- Word embeddings from Skip-gram model (256d).
- Sense embeddings from SCA model (256d).
- The concatenation of word embeddings and sense embeddings (512d).
- Word embeddings from Skip-gram model, which have the same dimension with the concatenation (512d).

For all experiments on downstream tasks, we perform significant test on word embedding (256d) versus sense embedding (256d) and concatenation (512d). Additionally, significant test is performed on word embedding with double dimension (512d) and concatenation (512d) to exclude the impact of increased dimensions of embedding.

6.1 Event Detection

Event detection aims to extract events with specific types from unstructured data. The ACE 2005[6] Chinese corpus is used for dataset, divided into training, validation and test set. We perform event detection as a sequence tagging task with a classic Bidirectional LSTM-CRF model [14]. We use early stop strategy and dropout on input and output of Bi-LSTM with rate 0.5. Adam method is applied for parameter optimization. The experimental results on event detection are shown in Table 4.

Table 4. Micro F1 on event trigger classification. P value <0.05 for concatenation versus word (256) and word (512), besides P value >0.05 for sense.

Embeddings	Micro F1
Word (256d)	62.38 ± 0.99
Sense (256d)	62.89 ± 0.45
Word (512d)	62.35 ± 0.69
Word $\oplus$ Sense (512d)	$\mathbf{63.78 \pm 1.19}$

6.2 Relation Classification

ACE 2005 RDC Chinese dataset contains 633 documents, 9,317 relation mentions tagged with 6 major relation types and 18 subtypes. We retain 8,489 instances whose distances between the two entity are less than 15 and split them into training , validation and test sets. Following Nguyen and Grishman [23], we use a single layer convolutional neural network with dropout on the input and output of convolution layer to predict the relation type of entity pairs. The training strategy is same as for the event detection task. The experimental results are shown in Table 5.

Table 5. Experiment results on relation classification. P value <0.01 for all significant test.

Embedding	Micro F1
Word (256d)	82.72 ± 0.94
Sense (256d)	84.20 ± 0.46
Word (512d)	82.91 ± 0.62
Word $\oplus$ Sense (512d)	$\mathbf{85.75 \pm 0.44}$

6.3 Text Classification

We use a subset sampling from THUCnews[7] as the dataset, which contains 65000 Chinese news documents tagged with 10 news topics. Each news document has an average of 529 words. We divided the dataset into training, validation and test sets. Following Kim [16], we train a convolutional neural network with our pre-trained embeddings. The training strategy is same as for the precious tasks. The experimental results on text classification are shown in Table 6.

Table 6. Experiment results on text classification. P value <0.05 for word (256d) versus sense (256d) and <0.01 for other significant tests.

Embedding	Accuracy
Word (256d)	95.84 ± 0.17
Sense (256d)	95.60 ± 0.20
Word (512d)	95.87 ± 0.09
Word $\oplus$ Sense (512d)	$\mathbf{96.35 \pm 0.13}$

[6] https://catalog.ldc.upenn.edu/LDC2006T06

[7] http://thuctc.thunlp.org/

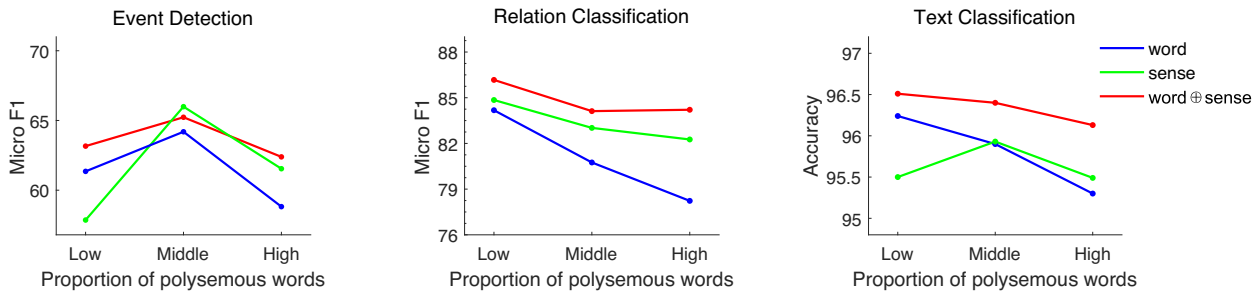

Figure 3. The performance of different embedding settings in three tasks. The test set of each downstream task is divided into three subsets with equal size according to the proportion of polysemous words.

6.4 Main Result of Extrinsic Evaluation

From Table 4 to Table 6, we find that sense embeddings outperform word embeddings in both event extraction and relation classification, but not in text classification. The concatenation of word embeddings and sense embeddings achieves significant improvement in all tasks, which indicates that both word and sense embeddings convey some complementary semantic information to each other and the integration of both two embeddings can benefit downstream tasks with more semantic information. In addition, comparing the results of word (256d) and word (512d), we can see that the increasing of dimension does not have a significant influence, which indicates that the improvement brought by concatenation is not due to the increased dimension, and extra semantic information from sense embeddings indeed plays an important role. In this sense, our sense embeddings can be used as a supplement to word embeddings, instead of a simple replacement.

6.5 Mitigation of the Impact of Polysemy

To further analyze the impact of polysemy in downstream tasks and explore how our sense embeddings can mitigate it, we divide the test set of each downstream task into three subsets with equal size but different proportions (low / middle / high) of polysemous words. The results on three subsets are shown in Figure 3.

For all tasks, the proportion of polysemous words affects the results. Overall, the performance for all tasks declines as the proportion of polysemous words increases, which indicates that the polysemous words indeed have a negative impact on downstream tasks. Particularly, the influence of polysemy has no obvious trend in event detection. The possible reason is that the largest proportion of token tags consists of "O" tag and whether they are polysemous may not effect the performance significantly.

Though the performance of our sense embeddings is ordinary when the polysemous proportion is Low, these sense embeddings outperform conventional word embeddings over three tasks when the polysemous proportion is Middle and High. The reason lies in that our sense embeddings are superior in representing the semantic information of polysemous words, making the downstream task more insensitive to the impact of polysemy.

The concatenation of word and sense embeddings can take the advantages of both representations and outperform each of them alone in three tasks regardless of the polysemous proportion. This proves the conclusion in Section 7 that our sense embeddings should be used as a supplement to word embeddings. Additionally, the gap between the concatenation and the word embeddings alone widens as the proportion increases, which indicates that the integration of sense embeddings can effectively mitigate the impact of polysemy.

6.6 Discussion on SCA Model and Contextualized Representations

We have demonstrated that sense representations generated by SCA model can be easily integrated into downstream tasks and gain improvement. However, we did notice that recent work in contextualized representations (e.g. BERT [6]) have achieved great performance on a lot of NLP tasks, benefiting from their complex models, large-scale corpus and long-range contextual information. Compared to these language models, our light-weighted model essentially focus on sense representation learning, which can build dynamic word representations by weighting sense representations during inference without additional parameters introduced. It's unfair to directly compare SCA model with language models, yet we believe it is a valuable future work to investigate how to incorporate language models with prior human knowledge of word senses. Note that the usage of language models means the context is always required, while sense embeddings can be also useful in context-free situations. For example, accurate representations of word senses is beneficial to the construction and upgrading of linguistic infrastructures like semantic dictionary and human knowledge base.

7 Conclusion

In this paper, we propose a knowledge-based method to learn individual sense representations which can effectively support downstream tasks. To achieve that, we proposed to leverage the semantic information in sememes to model the context distribution of each sense and perform a reliable soft word sense disambiguation step with a contextual attention mechanism. We evaluate the intrinsic quality of our sense embeddings by a word similarity task and qualitative investigations, which indicate the significant advantages of our method in capturing exact meanings of senses. Extrinsic experimental results and the further analysis demonstrate that our sense embeddings gains great improvements by mitigating the impact of polysemy when integrated into downstream NLP tasks. In the future, it will be valuable to explore the incorporation of language models and prior human knowledge of word senses to support NLP tasks.

Acknowledgements

We would like to thank Handan Institute of Innovation, Peking University for their support of our work.

REFERENCES

[1] Leonard Bloomfield, 'A set of postulates for the science of language', *Language*, **2**(3), 153–164, (1926).

[2] Jose Camacho-Collados and Mohammad Taher Pilehvar, 'From word to sense embeddings: A survey on vector representations of meaning', *Journal of Artificial Intelligence Research*, **63**, 743–788, (2018).

[3] Tao Chen, Ruifeng Xu, Yulan He, and Xuan Wang, 'Improving distributed representation of word sense via wordnet gloss composition and context clustering', in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, volume 2, pp. 15–20, (2015).

[4] Xinxiong Chen, Zhiyuan Liu, and Maosong Sun, 'A unified model for word sense representation and disambiguation', in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1025–1035, (2014).

[5] Liuling Dai, Bin Liu, Yuning Xia, and ShiKun Wu, 'Measuring semantic similarity between words using hownet', in *2008 International Conference on Computer Science and Information Technology*, pp. 601–605. IEEE, (2008).

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova, 'Bert: Pre-training of deep bidirectional transformers for language understanding', *arXiv preprint arXiv:1810.04805*, (2018).

[7] Zhendong Dong and Qiang Dong, 'Hownet-a hybrid language and knowledge resource', in *International Conference on Natural Language Processing and Knowledge Engineering, 2003. Proceedings. 2003*, pp. 820–824. IEEE, (2003).

[8] LI Feng and LI Fang, 'An new approach measuring semantic similarity in hownet 2000 [j]', *Journal of Chinese Information Processing*, **3**, 018, (2007).

[9] Bin Ge, Fang-Fang Li, Si-Lu Guo, and Da-Quan Tang, 'Word's semantic similarity computation method based on hownet', *Jisuanji Yingyong Yanjiu*, **27**(9), 3329–3333, (2010).

[10] Yihong Gu, Jun Yan, Hao Zhu, Zhiyuan Liu, Ruobing Xie, Maosong Sun, Fen Lin, and Leyu Lin, 'Language modeling with sparse product of sememe experts', in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 4642–4651, (2018).

[11] Yi Guan, Xiao-long Wang, Xiang-Yong Kong, and Jian Zhao, 'Quantifying semantic similarity of chinese words from hownet', in *Proceedings. International Conference on Machine Learning and Cybernetics*, volume 1, pp. 234–239. IEEE, (2002).

[12] Wenpeng Hu, Jiajun Zhang, and Nan Zheng, 'Different contexts lead to different word embeddings', in *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pp. 762–771, (2016).

[13] Eric H Huang, Richard Socher, Christopher D Manning, and Andrew Y Ng, 'Improving word representations via global context and multiple word prototypes', in *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pp. 873–882. Association for Computational Linguistics, (2012).

[14] Zhiheng Huang, Wei Xu, and Kai Yu, 'Bidirectional lstm-crf models for sequence tagging', *arXiv preprint arXiv:1508.01991*, (2015).

[15] Ignacio Iacobacci, Mohammad Taher Pilehvar, and Roberto Navigli, 'Sensembed: Learning sense embeddings for word and relational similarity', 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)*, volume 1, pp. 95–105, (2015).

[16] Yoon Kim, 'Convolutional neural networks for sentence classification', *arXiv preprint arXiv:1408.5882*, (2014).

[17] Guang-He Lee and Yun-Nung Chen, 'Muse: Modularizing unsupervised sense embeddings', in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 327–337, (2017).

[18] Jiwei Li and Dan Jurafsky, 'Do multi-sense embeddings improve natural language understanding?', in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 1722–1732, (2015).

[19] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean, 'Efficient estimation of word representations in vector space', *arXiv preprint arXiv:1301.3781*, (2013).

[20] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean, 'Distributed representations of words and phrases and their compositionality', in *Advances in neural information processing systems*, pp. 3111–3119, (2013).

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

[22] Arvind Neelakantan, Jeevan Shankar, Alexandre Passos, and Andrew McCallum, 'Efficient non-parametric estimation of multiple embeddings per word in vector space', *arXiv preprint arXiv:1504.06654*, (2015).

[23] Thien Huu Nguyen and Ralph Grishman, 'Relation extraction: Perspective from convolutional neural networks', in *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing*, pp. 39–48, (2015).

[24] Yilin Niu, Ruobing Xie, Zhiyuan Liu, and Maosong Sun, 'Improved word representation learning with sememes', in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pp. 2049–2058, (2017).

[25] Jeffrey Pennington, Richard Socher, and Christopher Manning, 'Glove: Global vectors for word representation', in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pp. 1532–1543, (2014).

[26] Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer, 'Deep contextualized word representations', *arXiv preprint arXiv:1802.05365*, (2018).

[27] Mohammad Taher Pilehvar and Nigel Collier, 'De-conflated semantic representations', *arXiv preprint arXiv:1608.01961*, (2016).

[28] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever, 'Improving language understanding by generative pre-training', *URL https://s3-us-west-2. amazonaws. com/openai-assets/researchcovers/languageunsupervised/language understanding paper. pdf*, (2018).

[29] Joseph Reisinger and Raymond J Mooney, 'Multi-prototype vector-space models of word meaning', in *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pp. 109–117. Association for Computational Linguistics, (2010).

[30] Sascha Rothe and Hinrich Schütze, 'Autoextend: Extending word embeddings to embeddings for synsets and lexemes', *arXiv preprint arXiv:1507.01127*, (2015).

[31] David E Rumelhart, Geoffrey E Hinton, Ronald J Williams, et al., 'Learning representations by back-propagating errors', *Cognitive modeling*, **5**(3), 1, (1988).

[32] Hinrich Schütze, 'Automatic word sense discrimination', *Computational linguistics*, **24**(1), 97–123, (1998).

[33] Amos Tversky and Itamar Gati, 'Similarity, separability, and the triangle inequality.', *Psychological review*, **89**(2), 123, (1982).

[34] Xuefeng Yang and Kezhi Mao, 'Learning multi-prototype word embedding from single-prototype word embedding with integrated knowledge', *Expert Systems with Applications*, **56**, 291–299, (2016).