Front. Comput. Sci., 2021, 15(5): 155327
https://doi.org/10.1007/s11704-020-0002-4

REVIEW ARTICLE

Sememe knowledge computation: a review of recent advances in application and expansion of sememe knowledge bases

Fanchao QI[1,2,3], **Ruobing XIE**[4], **Yuan ZANG**[1,2,3], **Zhiyuan LIU** (✉)[1,2,3,5], **Maosong SUN**[1,2,3,5]

1 Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China
2 Institute for Artificial Intelligence, Tsinghua University, Beijing 100084, China
3 Beijing National Research Center for Information Science and Technology, Beijing 100084, China
4 Search Product Center, WeChat Search Application Department, Tencent, Beijing 100080, China
5 Beijing Academy of Artificial Intelligence, Beijing 100191, China

Abstract A sememe is defined as the minimum semantic unit of languages in linguistics. Sememe knowledge bases are built by manually annotating sememes for words and phrases. HowNet is the most well-known sememe knowledge base. It has been extensively utilized in many natural language processing tasks in the era of statistical natural language processing and proven to be effective and helpful to understanding and using languages. In the era of deep learning, although data are thought to be of vital importance, there are some studies working on incorporating sememe knowledge bases like HowNet into neural network models to enhance system performance. Some successful attempts have been made in the tasks including word representation learning, language modeling, semantic composition, etc. In addition, considering the high cost of manual annotation and update for sememe knowledge bases, some work has tried to use machine learning methods to automatically predict sememes for words and phrases to expand sememe knowledge bases. Besides, some studies try to extend HowNet to other languages by automatically predicting sememes for words and phrases in a new language. In this paper, we summarize recent studies on application and expansion of sememe knowledge bases and point out some future directions of research on sememes.

Keywords natural language process, semantics, knowledge base, sememe, HowNet

1 Introduction

In the field of natural language processing (NLP), a hierarchical set of meaningful linguistic units are involved including documents, discourse, sentences, phrases and words, as shown in Fig. 1. Words are usually the most common processing units, mainly because a word is the smallest element of human languages that can stand by itself. But from the semantic perspective, a word may have multiple senses, and even a sense can be further split into smaller units (Fig. 1). For example, the most commonly used sense of the English word "boy" can be represented by the composition of meanings of "human", "male" and "child". In linguistics, a *sememe* is defined as the minimum semantic unit of human languages [1].

It is believed by some linguists that meanings of all the words in any language can be represented by a limited set of sememes (or semantic primitives) [2]. Theoretically, sememes are helpful in understanding and utilizing human languages better and deeper. However, sememes are implicit in text, and it is hard to determine and utilize sememes for ordinary people. To put the sememe theory into practice of NLP, sememe knowledge bases (KBs) are built which contain words and phrases manually annotated with sememes.

HowNet [3] is one of the most famous sememe KBs. It has an elaborate sememe-based annotating system which comprises about $2,000$ pre-defined sememes. Under the annotating system, senses of more than $100,000$ Chinese and English words are separately annotated with several sememes, where there are also relations between sememes. The sememes annotated to a sense actually form a "sememe tree", where different labels are attached to parent-child node pairs. The root of a sememe tree is a *categorial sememe* which depicts the main part of a sense [3].

Figure 2 illustrates an example of how words are annotated with sememes in HowNet. The English word "husband" has two senses, namely "married man" and "carefully use". The first sense is annotated with four sememes in HowNet, namely

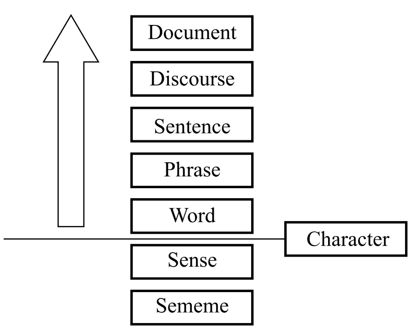


Fig. 1 Different linguistic units in NLP studies

Received January 1, 2020; accepted July 2, 2020

E-mail:liuzy@tsinghua.edu.cn

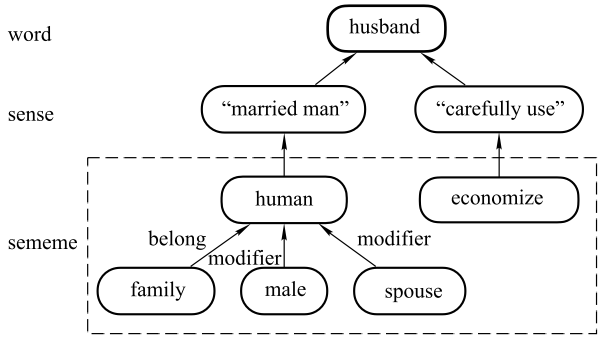


Fig. 2 An example of how words are annotated with sememes in HowNet

a categorical sememe `human` and three subsidiary sememes including `family`, `male` and `spouse`. The relations between the categorical sememe and three subsidiary sememes are belong, modifier and modifier, respectively. The second sense has only one sememe, i.e., the categorical sememe `economize`.

Since HowNet was published, it has attracted considerable research attention. Researchers have attempted to apply sememes to various NLP tasks involving information structure annotation [4], word similarity computing [5], word sense disambiguation [6, 7], sentiment analysis [8–10], text classification [11], etc. With more and more explorations, the effectiveness and power of sememes have gradually manifested themselves, and HowNet has become more and more influential in the field of NLP.

When research on NLP stepped into the era of deep learning, however, studies into linguistic KBs including HowNet began to fade away while data and corpora gradually stood on the center stage of NLP. It was believed that more complicated neural networks with more training data will always yield better performance. On the contrary, linguistic KBs including HowNet were thought to be difficultly compatible with neural models, and they were taken into less and less consideration.

In recent years, more and more studies have reconsidered the role of human knowledge played in artificial intelligence. There is some influential work which incorporates linguistic KBs into neural networks and obtains better system performance [12–14]. In terms of HowNet, some attempts have also been made to utilize it in neural network-based NLP. For example, word representation learning [15, 16], language modeling [17], lexicon expansion [18], semantic composition [19], sequence modeling [20], aspect extraction [21], reverse dictionary [22] and textual adversarial attack [23].

At the same time, some studies on sememe KB expansion have also been conducted. HowNet was entirely annotated by linguistic experts. Its construction took more than two decades. After its initial publication, the authors kept updating it by adding and revising sememe annotations of words and phrases. Manual annotation and revision of HowNet is very time-consuming and labor-intensive, and in fact, HowNet has stopped updating since 2014. In this Internet era, however, more and more new words and expressions are constantly emerging, and meanings of existing words keep changing rapidly, which render manual annotation and update even more difficult. To tackle this issue, Xie et al. [24] for the first time present the task of *sememe prediction*, which is aimed at automatically predicting sememes for unannotated words and expanding HowNet. Xie et al. [24] also propose two sememe prediction methods based on collaborative filtering [25] and matrix factorization [26] respectively. Following it, Jin et al. [27] incorporate Chinese characters into sememe prediction to obtain better prediction results, and Du et al. [28] utilize dictionary definitions to predict sememes for words and phrases.

Additionally, some work focuses on extending HowNet to other languages. HowNet contains Chinese and English words and phrases only. To our best knowledge, other languages have no sememe KBs like HowNet, which means NLP applications of these languages cannot benefit from sememe knowledge. To solve the problem, Qi et al. [29] propose the task of cross-lingual sememe prediction, aiming to automatically predict sememes for words in another language and gradually build a sememe KB for a new language. Qi et al. [30] further propose to build a multilingual sememe KB based on a multilingual encyclopedia dictionary, which can annotate sememes for words in many languages simultaneously and is more efficient and economical.

In this paper, we summarize recent NLP studies on sememes and HowNet to illustrate the advances in utilizing sememes in neural networks and automatically expanding sememe KBs. Specifically, we first introduce recent applications of sememe KBs in Section 2, then we describe some studies on monolingual and cross-lingual sememe prediction and sememe KB expansion in Section 3, and finally we draw a conclusion and point out some future research directions of sememe KBs in Section 4.

2 Application of sememe knowledge bases

Sememe KBs are quite special. Most linguistic KBs, such as WordNet [31] and ConceptNet [32], take words as elements and contain word-level relations. For example, WordNet groups synonyms into synsets and incorporates many semantic relations between synsets including "hypernym", "hyponym", "antonym", etc. Different from these linguistics KBs, HowNet are based on a limited number of sememes and focus on the compositional relations between words and the infra-word sememes. The particularity of HowNet leads to its unique strengths. First, the nature of HowNet that limited sememes can represent unlimited meanings makes it useful in low data regimes, and a typical use is improving word embeddings, especially for the low-frequency words [15, 16]. Second, it focuses on the semantic composition from sememes to words, which endows sememes with special suitability for integration into neural networks – words usually correspond to minimum processing units of neural network models, and it should be reasonable to incorporate sememes of a word into the processing unit of the word in some way [17, 19].

There have been some studies on utilizing sememes in neural network-based NLP, and almost all of them ignore the tree structures of sememes, instead, they regard sememes as semantic labels of words or senses of words. In the following, we first introduce two studies on using sememes to improve word representation learning, then we detail two pieces of work which utilize sememes in language modeling and semantic composition respectively, and finally we briefly describe some other studies into application of sememes.

2.1 Sememe-guided word representation learning

Recently, unsupervised word representation learning models, especially CBOW and Skip-gram of `word2vec` [33], have brought revolutionary change to NLP research. Some work has tried to incorporate word-level linguistic KBs into word representation learning to obtain better word embeddings [13, 32]. The attempts to use sememes KBs in word representation learning have also been made. Considering the difference between HowNet and other word-level KBs, their incorporation ways are also different.

Next, we elaborate on two representative studies on using sememes in word representation learning. But before that, we first give a brief introduction to the two classical word representation learning models CBOW and Skip-gram [33], on which the two studies are based.

2.1.1 CBOW and Skip-gram

The ideas of both CBOW and Skip-gram assume that the meaning of a word is highly related to its contexts. The difference is that CBOW supposes contexts should predict the target word while Skip-gram uses the target word to predict its contexts. Formally, for a word sequence $H = \{w_0, \dots, w_N\}$, CBOW intends to maximize the predictive probability:

$$\mathcal{L}^{CBOW}(H) = \sum_{n=K}^{N-K} \log P(w_n|w_{n-K}, \dots, w_{n-1}, w_{n+1}, \dots, w_{n+K}), \tag{1}$$

where K is the size of context sliding window, and

$$P(w_n|w_{n-K}, \dots, w_{n-1}, w_{n+1}, \dots, w_{n+K}) = \frac{\exp(\mathbf{w}_c^{\mathrm{T}}\mathbf{w}_n)}{\sum_{w'\in\mathbb{W}} \exp(\mathbf{w}_c^{\mathrm{T}}\mathbf{w}')}, \tag{2}$$

where $\mathbb{W}$ is the vocabulary and $\mathbf{w}_c$ represents the context vector:

$$\mathbf{w}_c = \frac{1}{2K} \sum_{k=-K,\ k\neq 0}^{K} \mathbf{w}_{n+k}. \tag{3}$$

Similarly, Skip-gram maximizes the following predictive probability:

$$\mathcal{L}^{SG}(H) = \sum_{n=K}^{N-K} \log P(w_{n-K}, \dots, w_{n-1}, w_{n+1}, \dots, w_{n+K}|w_n), \tag{4}$$

where

$$P(w_{n-K}, \dots, w_{n-1}, w_{n+1}, \dots, w_{n+K}|w_n) = \prod_{k=-K,\ k\neq 0}^{K} P(w_{n+k}|w_n),$$
$$P(w_{n+k}|w_n) = \frac{\exp(\mathbf{w}_{n+k}^{\mathrm{T}}\mathbf{w}_n)}{\sum_{w'\in\mathbb{W}} \exp(\mathbf{w}_{n+k}^{\mathrm{T}}\mathbf{w}')}. \tag{5}$$

To accelerate the calculation of softmax functions, hierarchical softmax and negative sampling are proposed [33].

2.1.2 Post-processing word embeddings with sememes

We first introduce the first work which takes account of sememes in word representation learning [15]. In this work, word embeddings are first learned by `word2vec` or any other methods, then sememe embeddings are learned using word embeddings, and finally word embeddings are substituted or enhanced with sememe embeddings.

Specifically, sememe embeddings are learned using a similar way to CBOW, i.e., sememes of a word should predict the word. Formally, the following predictive probability is maximized during training:

$$\mathcal{L}^{Sememe}(H) = \sum_{n=0}^{N} \log P(w_n|\mathbb{X}(w_n)), \tag{6}$$

where $\mathbb{X}(w_n)$ is the sememe set of word w_n, and $P(w_n|\mathbb{X}(w_n))$ is defined as

$$P(w_n|\mathbb{X}(w_n)) = \frac{\exp(\bar{\mathbf{x}}(w_n)^{\mathrm{T}}\mathbf{w}_n)}{\sum_{w'\in\mathbb{W}} \exp(\bar{\mathbf{x}}(w_n)^{\mathrm{T}}\mathbf{w}')}, \tag{7}$$

where $\bar{\mathbf{x}}(w_n)$ is the average embedding of w_n's sememes:

$$\bar{\mathbf{x}}(w_n) = \frac{1}{|\mathbb{X}(w_n)|} \sum_{x\in\mathbb{X}(w_n)} \mathbf{x}, \tag{8}$$

where $|\cdot|$ represents the element number of a set. After obtaining sememe embeddings, embeddings of some words, e.g., the low-frequency words, can be substituted by the average of their sememe embeddings:

$$\mathbf{w}'_n = \bar{\mathbf{x}}(w_n). \tag{9}$$

In the above-mentioned method, polysemy is disregarded, and the sememe set of a word is the union of the sememe sets of all its senses. Another sense-aware sememe embedding learning method is also proposed. Suppose the sense set of w_n is $\mathbb{S}(w_n) = \{s_1^n, \dots, s_{|\mathbb{S}(w_n)|}^n\}$, and the sememe set of the i-th sense is $\mathbb{X}(s_i^n)$, the maximized predictive probability is altered to

$$\mathcal{L}^{Sememe}(H) = \sum_{n=0}^{N} \log P(w_n|s_r^n), \tag{10}$$

where $r = \arg\max_i \cos(\mathbf{s}_i^n, \mathbf{w}_c)$, i.e., the sense which is most semantically similar to the context, and embedding of a sense is defined as the average of its sememe embeddings:

$$\mathbf{s}_i^n = \frac{1}{|\mathbb{X}(s_i^n)|} \sum_{x\in\mathbb{X}(s_i^n)} \mathbf{x}. \tag{11}$$

2.1.3 Injecting sememes when learning word embeddings

The previous method separate the learning processes of word and sememe embeddings, where word embeddings are fixed during training sememe embeddings. Niu et al. [16] propose to incorporate sememes during the learning process of word embeddings, in which way word embeddings will have been enhanced after training with a corpus. They actually present three different models based on Skip-gram.

The first model is the simple sememe aggregation model (SSA). In this model, polysemy is also disregarded, and the target word embedding is substituted by the average embedding of its sememes in Eq. (5). Specifically,

$$P(w_{n+k}|w_n) = \frac{\exp(\mathbf{w}_{n+k}^{\mathrm{T}}\bar{\mathbf{x}}(w_n))}{\sum_{w'\in\mathbb{W}} \exp(\mathbf{w}_{n+k}^{\mathrm{T}}\mathbf{w}')}, \tag{12}$$

The second model is the sememe attention over context model (SAC). In this model, embedding of each context word, namely $\mathbf{w}_{n+k}$ in Eq. (5), is substituted by the weighted sum of embeddings of its sememes, where the attention mechanism is used to determine the weight of each sense. Formally,

$$\mathbf{w}_{n+k} = \sum_{i=1}^{|\mathbb{S}(w_{n+k})|} a(s_i^{n+k}) \cdot \hat{\mathbf{s}}_i^{n+k}, \tag{13}$$

where $\hat{\mathbf{s}}_i^{n+k}$ is a learnable vector representing another embedding of the sense s_i^{n+k}, $a(s_i^{n+k})$ is the attention weight calculated by a softmax function:

$$a(s_i^{n+k}) = \frac{\exp(\mathbf{w}_n^{\mathrm{T}} \mathbf{s}_i^{n+k})}{\sum_{i'=1}^{|\mathbb{S}(w_{n+k})|} \exp(\mathbf{w}_n^{\mathrm{T}} \mathbf{s}_{i'}^{n+k})}. \tag{14}$$

The third model is named the sememe attention over target model (SAT). This model adaptively selects the the most appropriate sense of the target word. The target word embedding, namely $\mathbf{w}_n$ in Eq. (5), is substituted by the weighted sum of embeddings of its sememes in a similar way to SAC. Formally,

$$\begin{aligned} \mathbf{w}_n &= \sum_{i=1}^{|\mathbb{S}(w_n)|} a(s_i^n) \cdot \hat{\mathbf{s}}_i^n, \\ a(s_i^n) &= \frac{\exp(\mathbf{w}_c^{\mathrm{T}} \mathbf{s}_i^n)}{\sum_{i'=1}^{|\mathbb{S}(w_n)|} \cdot \exp(\mathbf{w}_c^{\mathrm{T}} \mathbf{s}_{i'}^n)}, \end{aligned} \tag{15}$$

where $\mathbf{w}_c$ is still the context embedding defined in Eq. (3).

Empirical experiments have demonstrated that both methods, i.e., post-processing word embeddings with sememes and injecting sememes when learning word embeddings, improve the word embedding quality [15, 16]. Moreover, both methods obtain sense embeddings which can be used in word sense disambiguation and more precise representation of words in context.

2.2 Sememe-guided neural language modeling

Language modeling (LM) is aimed at predicting the next word given previous context and measuring the likelihood of a word sequence. LM is essential to understanding human languages and has been extensively studied. Recently, neural language models (NLMs), mostly based on recurrent neural networks (RNNs), have profoundly promoted human language understanding of machines. They use RNNs to encode the previous text into a vector and feed the vector to a classifier to predict the next word in decoding. Gu et al. [17] propose a sememe-driven language model (SDLM). Instead directly predicting the next word in the decoding phase, SDLM predicts sememes first, then predict senses and finally predict the next word.

In the following, we first briefly introduce the general framework of RNN-based NLMs, and then we elaborate the SDLM.

2.2.1 General framework of RNN-based NLMs

Formally, still for the word sequence $H = \{w_0, \ldots, w_N\}$, when predicting n-th word, the predictive probability is

$$P(w_n|w_0, \ldots, w_{n-1}) = \frac{\exp(\mathbf{h}_{n-1}^{\mathrm{T}} \mathbf{w_n})}{\sum_{w' \in \mathbb{W}} \exp(\mathbf{h}_{n-1}^{\mathrm{T}} \mathbf{w}')}, \tag{16}$$

where $\mathbf{h}_{n-1}$ is the $(n-1)$-th hidden state, which carries the semantic information of previous $n-1$ words. It can be calculated by

$$\mathbf{h}_{n-1} = \mathrm{RNN}(\mathbf{w}_{n-1}, \mathbf{h}_{n-2}, \mathbf{\Xi}_{n-1}; \mathbf{\Theta}), \tag{17}$$

where $\mathbf{\Xi}_{n-1}$ represents other state variables corresponding to the $(n-1)$-th word, e.g., cell states of LSTM [34], and $\mathbf{\Theta}$ signifies the learnable parameters of the RNN. The optimization objective is to maximize the following predictive probability:

$$\mathcal{L}(\mathbf{\Theta}) = \frac{1}{N} \sum_{n=0}^{N} \log P(w_n|w_0, \ldots, w_{n-1}). \tag{18}$$

2.2.2 Sememe-driven language model (SDLM)

In general NLM, the predictive probability of the n-th word is directly calculated using the previous hidden state and the n-th word's embedding, as shown in Eq. (16). In contrast, SDLM predicts sememes first, then senses, and finally words. Figure 3 gives an example showing how the next word is predicted in SDLM.

First, the sememes of the n-th word are predicted. The predictive probability of the sememe x_j is

$$q_j = P(x_j|w_0, \ldots, w_{n-1}) = \sigma(\mathbf{h}_{n-1}^{\mathrm{T}} \mathbf{v}_j + b_j), \tag{19}$$

where σ is the sigmoid function, $\mathbf{v}_j$ is a learnable parameter vector and b_j is is learnable scalar parameter.

Then the sense of the n-th word is predicted. Here a sememe is regarded as an "expert" making predictions of senses, following the theory of product of experts (PoE) [35]. Formally, the predictive probability of the sense s_i given its sememe x_j, is

$$P^{(x_j)}(s_i|w_0, \ldots, w_{n-1}) = \frac{\exp(q_j C_{i,j} \phi^{(x_j)}(\mathbf{h}_{n-1}, \mathbf{s}_i))}{\sum_{s_{i'} \in \hat{\mathbb{S}}(x_j)} \exp(q_j C_{i',j} \phi^{(x_j)}(\mathbf{h}_{n-1}, \mathbf{s}_{i'}))}, \tag{20}$$

where $\hat{\mathbb{S}}(x_j)$ is the set of senses annotated by the sememe x_j, $C_{i,j}$ is a normalization constant, $\phi^{(x_j)}(\cdot,\cdot)$ is a bilinear function parameterized with a parameter matrix $\mathbf{U}_j$:

$$\phi^{(x_j)}(\mathbf{h}_{n-1}, \mathbf{s}_i) = \mathbf{h}_{n-1}^{\mathrm{T}} \mathbf{U}_j \mathbf{s}_i. \tag{21}$$

Then the predictive probability of the sense s_i with all its sememes as experts is as follows:

$$\begin{aligned} P(s_i|w_0, \ldots, w_{n-1}) &\propto \prod_{x_j \in \mathbb{X}(s_i)} P^{(x_j)}(s_i|w_0, \ldots, w_{n-1}) \\ &= \frac{\exp(\sum_{x_j \in \mathbb{X}(s_i)} q_j C_{i,j} \phi^{(x_j)}(\mathbf{h}_{n-1}, \mathbf{s}_i))}{\sum_{s_{i'} \in \hat{\mathbb{S}}(x_j)} \exp(\sum_{x_j \in \mathbb{X}(s_i)} q_j C_{i',j} \phi^{(x_j)}(\mathbf{h}_{n-1}, \mathbf{s}_{i'}))} \\ &= \frac{\exp(\sum_{x_j \in \mathbb{X}(s_i)} q_j C_{i,j} \mathbf{h}_{n-1}^{\mathrm{T}} \mathbf{U}_j \mathbf{s}_i)}{\sum_{s_{i'} \in \hat{\mathbb{S}}(x_j)} \exp(\sum_{x_j \in \mathbb{X}(s_i)} q_j C_{i',j} \mathbf{h}_{n-1}^{\mathrm{T}} \mathbf{U}_j \mathbf{s}_{i'})}. \end{aligned} \tag{22}$$

Finally we simply sum up the predictive probabilities of senses of the nth word to obtain the word predictive probability:

$$P(w_n|w_0, \ldots, w_{n-1}) = \sum_{s_i \in \mathbb{S}(w_n)} P(s_i|w_0, \ldots, w_{n-1}). \tag{23}$$

SDLM is highly adaptable. It can be combined with any other NLMs with different encoding architectures. In fact, it can also

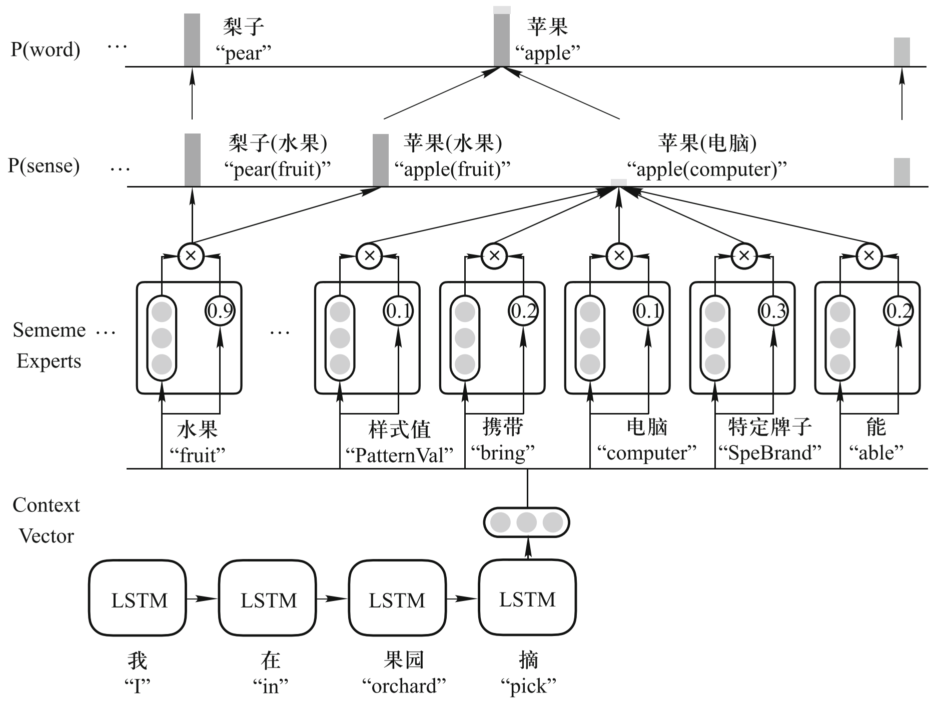


Fig. 3 An example illustrating the architecture of SDLM

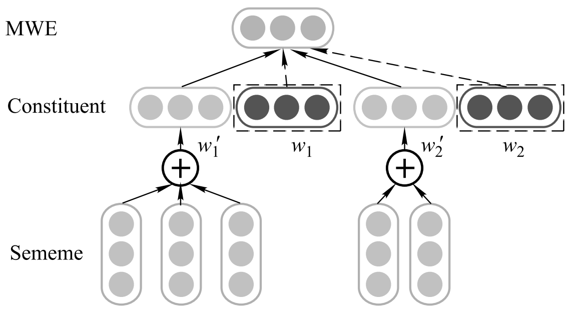


Fig. 4 The SCAS model

be used in the decoding phase of generative models. Empirical experiments have been conducted using popular LSTM-based NLMs. Experimental results demonstrate that SDLM can improve the performance of existing NLMs and can also bring enhancement of generative text quality in a headline generation task [17].

2.3 Sememe-guided semantic composition modeling

Semantic composition (SC) refers to the phenomenon that the meaning of a syntactically complex unit is a function of meanings of the complex unit's constituents and the combination rule [36]. SC is regarded as the the fundamental truth of semantics [37] by some linguists. In the field of NLP, SC has been explored in various tasks inlcuding language modeling [38], syntactic parsing [39] and sentiment analysis [40, 41].

The most widely studied topic in SC is the composition from words to multi-word expression (MWEs), more specifically, the distributional vector representations of MWEs. Mitchell et al. [42] propose a general framework to model MWE representation learning:

$$\mathbf{p} = f(\mathbf{w}_1, \mathbf{w}_2, R, K), \tag{24}$$

where f is the composition function, $\mathbf{p}$ denotes the embedding of an MWE, $\mathbf{w}_1$ and $\mathbf{w}_2$ represent the embeddings of the MWE's two constituents, R stands for the combination rule, and K denotes external knowledge required to construct the semantics of the MWE. Notice that this formula is for two-word MWEs but can be easily extended to longer MWEs.

Almost all existing work focuses on designing a more complicated composition function f, some takes the the combination rule R into consideration, and no previous work attempts to use external knowledge K.

Qi et al. [19] makes the first attempt to incorporate external linguistics knowledge, i.e., sememes, into modeling SC. They propose two sememe-incorporated MWE representation learning models. The first model is the semantic compositionality with aggregated sememe model (SCAS), which simply concatenates the embeddings of the MWE's constituents and their sememes to obtain the MWE's embedding. The second is the semantic compositionality with mutual sememe attention model (SCMSA), which considers the mutual attention between a constituent's sememes and the other constituent. Figs. 4 and 5 illustrate the architectures of the two models respectively.

2.3.1 The SCAS model

Formally, for an MWE $p = \{w_1, w_2\}$, its embedding can be obtained by

$$\mathbf{p} = \tanh(\mathbf{W}_c^r[\mathbf{w}_1 + \mathbf{w}_2; \mathbf{w}_1' + \mathbf{w}_2'] + \mathbf{b}_c), \tag{25}$$

where $\mathbf{W}_c^r$ is the combination rule-related composition matrix (r represents a syntactic combination rule, e.g., adjective-noun), $\mathbf{b}_c$ is a learnable pamaemter vector, $\mathbf{w}_1'$ and $\mathbf{w}_2'$ represent the aggregated sememe embeddings of w_1 and w_2 respectively:

$$\mathbf{w}_1' = \sum_{x_j \in \mathbb{X}(w_1)} \mathbf{x}_j, \quad \mathbf{w}_2' = \sum_{x_j \in \mathbb{X}(w_2)} \mathbf{x}_j. \tag{26}$$

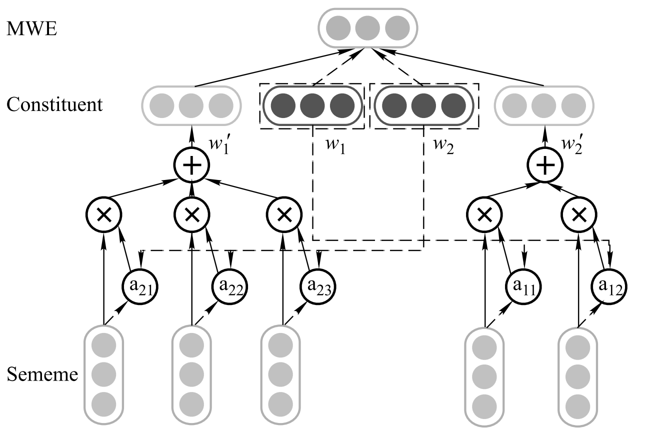


Fig. 5 The SCMSA model

2.3.2 The SCMSA model

A constituent may have multiple senses, and its meaning may vary with the other constituent in an MWE. Accordingly, the sememes of a constituent should have varying weights when the constituent is combined with different constituents. Qi et al. [19] propose the SCMSA model which employs the mutual attention mechanism to dynamically adjust weights of sememes.

Formally, w_2's sememe embedding with attention can be obtained by

$$\begin{aligned}\mathbf{e}_1 &= \tanh(\mathbf{W}_a\mathbf{w}_1 + \mathbf{b}_a),\\ a_{2,j} &= \frac{\exp(\mathbf{x}_j^{\mathrm{T}}\mathbf{e}_1)}{\sum_{x_{j'}\in\mathbb{X}(w_2)}\exp(\mathbf{x}_{j'}^{\mathrm{T}}\mathbf{e}_1)},\\ \mathbf{w}_2' &= \sum_{x_j\in\mathbb{X}(w_2)} a_{2,j}\mathbf{x}_j,\end{aligned}\tag{27}$$

where $\mathbf{W}_a$ is the weight matrix and $\mathbf{b}_a$ is a learnable parameter vector. $\mathbf{w}_1'$ can be calculated similarly. Finally, the embedding of the MWE can be obtained by Eq. (25).

Extensive intrinsic and extrinsic evaluations verify the effectivenss of sememes in MWE representation learning [19]. They also find that sememes are useful in measuring SC degree.

2.4 Other applications of sememe knowledge bases

In addition to the above-mentioned studies on utilizing sememe KBs, more explorations of incorporating sememe KBs into neural models have been conducted. Luo et al. [21] use sememes in unsupervised neural aspect extraction. They regard sememe KBs as external semantic resources to enhance sentence representation. Specifically, they incorporate sememes in a sentence encoder via a hierarchical and a context-enhanced attention mechanisms respectively, and then learn sentiment aspects by latent variables when decoding. Qin et al. [20] also utilize sememes to enhance sentence representation learning. But they directly inject sememes into RNN cells by three different methods and improve the sequence modeling ability of RNNs. Zang et al. [23] take advantage of the nature of sememe KBs that words with the same sememe annotations are semantically identical. They substitute the original words in a sentence with the words with the same sememe annotations to generate adversarial examples and use them to attack neural models. Experimental results also demonstrate that their sememe-based word substitution is better than the substitution methods based on other KBs like WordNet in terms of adversarial attacking.

3 Expansion of sememe knowledge bases

Although a large number of words and phrases has been absorbed into HowNet (127,266 for Chinese, 118,263 for English), more and more new words and expressions are emerging in this Internet era. Meanwhile, the meanings of existing words and expressions keep changing. Therefore, continuous update, including expansion and revision, is necessary for HowNet and other lexical linguistic KBs. Before 2014, HowNet was regularly updated manually by its authors, although it was a little strenuous. But after 2014, authors of HowNet has stopped updating it, which makes HowNet experience difficulty when being used in processing latest text. To tackle this issue, Xie et al. [24] propose to automatically predict sememes for unannotated words, which can assist in human annotation in terms of both efficiency and accuracy. Further, Jin et al. [27] incorporate Chinese character information into sememe prediction to improve accuracy of sememe prediction. For another thing, HowNet covers only two languages, and there are no sememe KBs like HowNet for other languages. To solve the problem, Qi et al. [29] for the first time present the task of *cross-lingual lexical sememe prediction*, aiming to automatically predict sememes for words in another language. Qi et al. [30] propose a more efficient and economical method in which a multilingual sememe KB containing annotated words in many languages will be built based on BabelNet [43], a multilingual encyclopedia dictionary. Notice that all these studies regard sememes as semantic labels of words or phrases, and they only predict unstructured sememes while disregarding the hierarchical structures and relations of sememes.

Next, we first introduce the two studies on monolingual sememe prediction [24, 27], then we present the two cross- or multi-lingual sememe prediction studies.

3.1 Monolingual sememe prediction

3.1.1 Sememe prediction based on collaborative filtering and matrix factorization

In this work, sememe prediction is actually modeled as a recommendation task, where words/phrases are regarded as users and sememes are regarded as products. Predicting sememes for a word is to recommend products to a new user. They adopt two classical method of recommendation system, namely collaborative filtering [25] and matrix factorization [26].

The basic idea of collaborative filtering is to recommend similar products to similar users. The first model, sememe prediction with word embeddings (SPWE) is based on the idea and predict similar sememes for similar words. They use word embeddings to measure the similarity between words. Formally, they calculate the prediction score of each sememe for the given target word w:

$$P(x_j|w) = \sum_{w_i'\in\mathbb{W}'} \cos(\mathbf{w}, \mathbf{w}_i') \cdot \mathbf{M}_{ij} \cdot c^{r_i},\tag{28}$$

where $\mathbb{W}'$ is the set of words with known sememe annotations, $\mathbf{M}$ is a binary matrix denoting the annotation relations between words and sememes, i.e., if the word w_i is annotated by the sememe x_j, $\mathbf{M}_{ij} = 1$, otherwise $\mathbf{M}_{ij} = 0$. To avoid noisy sememes brought by the semantically irrelevant words, an extra confidence factor c^{r_i} is introduced, where $c \in (0, 1)$ is a hyper-parameter, r_i is the descending rank of the word similarity $\cos(\mathbf{w}, \mathbf{w}'_i)$. By doing this, the semantically irrelevant words will have little influence on sememe prediction.

As for the matrix factorization-based sememe prediction with sememe embeddings model (SPSE), its idea is to learn sememe embedding in the same distributional semantic space as words by decomposing the word-sememe annotation matrix $\mathbf{M}$, and then predict sememes with close embeddings to the target words. Specifically, a set of pre-trained word embeddings are used, and they are fixed to train sememe embeddings using the following loss function:

$$\mathcal{L} = \sum_{w_i \in \mathbb{W}', x_j \in \mathbb{X}} (\mathbf{w}_i^{\mathrm{T}} \mathbf{x}_j + b_i + b'_j - \mathbf{M}_{ij})^2, \tag{29}$$

where $\mathbf{b}_i$ and $\mathbf{b}'_j$ are two scalar parameters. After obtaining sememe embeddings, the prediction score of a sememe is proportional to the cosine similarity between its embedding and the target word's embedding:

$$P(x_j|w) \propto \cos(\mathbf{w}, \mathbf{x}_j). \tag{30}$$

The two models can combined to an ensemble model simply by totaling their sememe prediction scores up. The final sememe prediction result for a target word is the sememes with prediction scores higher than a threshold.

3.1.2 Incorporating Chinese characters into sememe prediction

The previous method uses word embeddings to represent the semantics of words, which only carry the external semantic information for words, i.e., contexts. In fact, internal information is complementary to external information and also useful in capturing word semantics, especially for the low-frequency words with little available external information. In this work, Jin et al. [27] attempt to utilize internal information of words, namely Chinese characters, to conduct sememe prediction.

They also propose two models, the first is the sememe prediction with word-to-character filtering model (SPWCF). The idea of this model is essentially collaborative filtering, too. Different from SPWE, it uses the Chinese character-based feature to measure similarity between words. Specifically, it considers words as similar if they contain the same characters at the same positions. Three positions are included, i.e., `BEGIN` (corresponding to the first character in a word), `END` (corresponding to the last character in a word), and `MIDDLE` (corresponding to the other characters in a word).

Formally, for a target word $w = c_1 c_2 \ldots c_{|w|}$, the character-based sememe prediction score is first calculated according to statistics of all the annotated words, which is defined as:

$$P(x_j|c, p) \propto \frac{\sum_{w_i \in \mathbb{W}' \wedge c \in \pi_p(w_i)} \mathbf{M}_{ij}}{\sum_{w_i \in \mathbb{W}' \wedge c \in \pi_p(w_i)} |\mathbb{X}(w_i)|}, \tag{31}$$

where $p \in \{\mathtt{B}, \mathtt{E}, \mathtt{M}\}$ refers to a position, $\pi_p(w)$ denotes the set of characters at the position p of the word w, e.g., $\pi_{\mathtt{B}}(w) = \{c_1\}$, $\pi_{\mathtt{E}}(w) = \{c_{|w|}\}$ and $\pi_{\mathtt{M}}(w) = \{c_2, \ldots, c_{|w|-1}\}$. Then the sememe prediction score of a word can be obtained by aggregating the character-based sememe prediction scores:

$$P(x_j|w) \propto \sum_{p \in \{\mathtt{B},\mathtt{E},\mathtt{M}\}} \sum_{c \in \pi_p(w)} P(x_j|c, p). \tag{32}$$

SPWCF is simple and effective, but it ignores the relations between sememes, just like SPWE. Following SPSE, Jin et al. [27] propose the sememe prediction with character and sememe embeddings model (SPCSE). It also decomposes the word-sememe annotation matrix to learn sememe embeddings, but it uses the most representative character embeddings rather than word embeddings, considering low-frequency words usually have poor word embeddings. Specifically, it uses the pre-trained character embeddings from [44], which provides each character with multiple embeddings $\mathbf{c}^1, \ldots, \mathbf{c}^{N(c)}$. It adaptively selects the embedding of a character of a word which is closest to current sememe embedding as the the most representative character embedding and uses it to substitute the word embedding. Formally, given the target word $w = c_1 c_2 \ldots c_{|w|}$ and a sememe x_j, the most representative character embedding is picked by:

$$\hat{k}, \hat{r} = \arg\min_{k,r} [1 - \cos(\mathbf{c}_k^r, \mathbf{x}_j)], \tag{33}$$

where $\hat{k}$ and $\hat{r}$ represent the indices of character and its embedding of the most representative character embedding of a word, respectively. Then the most representative character embeddings are used in decomposing the word-sememe annotation matrix:

$$\mathcal{L} = \sum_{w_i \in \mathbb{W}', x_j \in \mathbb{X}} (\mathbf{c}_{\hat{k}}^{\hat{r}} \cdot \mathbf{x}_j + b_{\hat{k}}^c + b'_j - \mathbf{M}_{ij})^2, \tag{34}$$

where $b_{\hat{k}}^c$ and b'_j are two scalar parameters. After obtaining the sememe embeddings, the prediction score of a sememe x_j for the target word w is

$$P(x_j|w) \propto \cos(\mathbf{c}_{\hat{k}}^{\hat{r}}, \mathbf{x}_j). \tag{35}$$

SPWCF and SPCSE can be combined, and then further combined with the ensemble of SPWE and SPSE, because all of them utilize different information. As illustrated in Fig. 6, for high-frequency words with good word embeddings, the two models using external semantic information SPWE and SPSE are used, while for the low-frequency words with poor word embeddings, all the four models are employed to better predict sememes.

3.2 Cross-lingual sememe prediction

The fact that HowNet covers only two languages restricts its wide application, and also prevents better understanding and utilizing the languages without sememe KBs. Qi et al. [29] for the first time propose the task of Cross-lingual Lexical Sememe Prediction (CLSP) to tackle this issue, aiming to transfer the sememe annotations from a language with sememe annotations (the source language) to another language without sememe annotations (the target language). However, CLSP can conduct sememe predictions for one language at one time, which is inefficient when coping with multiple languages. To solve the problem, Qi et al. [30] put forward another way, i.e., building a multilingual sememe KB based on an existing multilingual encyclopedia dictionary BabelNet [43], which is much more

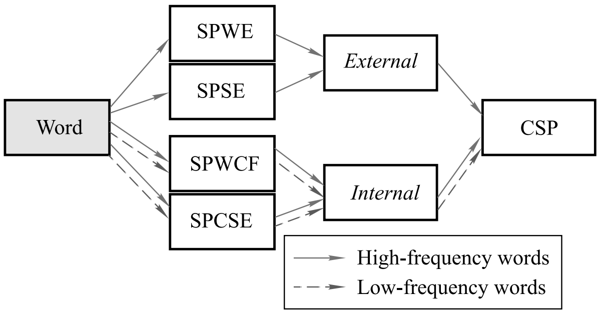


Fig. 6 The ensemble sememe prediction model

efficient and economical. Next we detail the two methods one by one.

3.2.1 Cross-lingual lexical sememe prediction

Qi et al. [29] propose a two-step cross-lingual lexical sememe prediction method. The first step is bilingual word representation learning, which is composed of three modules including (1) monolingual word representation learning, which is intended for learning embeddings of words in source and target languages respectively; (2) cross-lingual word embedding alignment, which is aimed at aligning the bilingual word embeddings into a unified semantic space; and (3) sememe knowledge incorporation whose objective is to incorporate sememes of the words in the source language into word representations. The second step is sememe prediction based on bilingual word embeddings in a unified semantic space, which is similar to SPWE.

The training loss of the first step has three corresponding parts:

$$\mathcal{L} = \mathcal{L}_{mono} + \mathcal{L}_{cross} + \mathcal{L}_{sememe}. \tag{36}$$

For the monolingual word representation learning part $\mathcal{L}_{mono}$, it comprises two independent subparts for the source and target languages respectively:

$$\mathcal{L}_{mono} = \mathcal{L}^S_{mono} + \mathcal{L}^T_{mono}. \tag{37}$$

For either language, we use Skip-gram to learn monolingual word embeddings using a monolingual corpus. Therefore, both $\mathcal{L}^S_{mono}$ and $\mathcal{L}^T_{mono}$ are in the similar form to Eq. (4).

The second part of the training loss, $\mathcal{L}_{cross}$, is also composed of two subparts:

$$\mathcal{L}_{cross} = \lambda_s \mathcal{L}_{seed} + \lambda_m \mathcal{L}_{match}, \tag{38}$$

where λ_s and λ_m are hyperparameters for controlling relative weightings of the two terms, $\mathcal{L}_{seed}$ and $\mathcal{L}_{match}$ correspond to alignment by seed lexicon and by self-matching respectively. $\mathcal{L}_{seed}$ encourages word embeddings of translation pairs in a seed lexicon $\mathbb{D}$ to be close, which is achieved via a L_2 regularizer:

$$\mathcal{L}_{seed} = \sum_{\langle w^S_s, w^T_t \rangle \in \mathbb{D}} \|\mathbf{w}^S_s - \mathbf{w}^T_t\|^2, \tag{39}$$

where w^S_s and w^T_t indicate a pair of words in source and target languages in the seed lexicon respectively. The self-matching is based on the assumption that each word in the target language (target word) should be matched to a single word in the source language (source word) or a special empty word, and vice versa.

The goal of self-matching is to find the matched source (target) word for each target (source) word and maximize the matching probabilities for all the matched word pairs. The loss of this part can be formulated as:

$$\mathcal{L}_{match} = \mathcal{L}^{T2S}_{match} + \mathcal{L}^{S2T}_{match}, \tag{40}$$

where $\mathcal{L}^{T2S}_{match}$ is the term for target-to-source matching and $\mathcal{L}^{S2T}_{match}$ is for source-to-target matching. Next, we detail the target-to-source matching loss $\mathcal{L}^{T2S}_{match}$, and the source-to-target matching is similar. Suppose m_t represents the index of the source word that the target word w^T_t matches with. $m_t = 0$ signifies the matched source word is the special empty word. $\mathcal{L}^{T2S}_{match}$ is defined as

$$\mathcal{L}^{T2S}_{match} = -\log P(C^T|C^S) = -\log \sum_{\mathbf{m}} P(C^T, \mathbf{m}|C^S), \tag{41}$$

where C^T and C^S denote the corpora in the target and source languages respectively, $\mathbf{m} = \{m_1, m_2, \ldots, m_{|\mathbb{W}^T|}\}$ ($\mathbb{W}^T$ is the vocabulary of the target language). We assume that the matching processes of target words are independent of each other. Therefore,

$$P(C^T, \mathbf{m}|C^S) = \prod_{w^T \in C^T} P(w^T, \mathbf{m}|C^S) = \prod_{t=1}^{|\mathbb{W}^T|} P(w^T_t|w^S_{m_t})^{c(w^T_t)}, \tag{42}$$

where $w^S_{m_t}$ is the source word that w^T_t matches with, and $c(w^T_t)$ is the number of times w^T_t occurs in the target corpus.

The third module of the bilingual word representation learning is to incorporate sememes of source words to improve word embeddings. Two different incorporation approaches are proposed. The first approach is straightforward, which makes embeddings of source words with similar sememe annotations close. This approach actually introduces semantically similar relations between words with the help of sememes, and hence it is named word relation-based approach. They group the source words sharing a certain number of sememes and force the embeddings of the words in a group to be closer in a similar way to [13]. Specifically, let $\hat{\mathbf{w}}^S_i$ be the adjusted word embedding of the source word w^S_i and $\mathrm{Syn}(w^S_i)$ denote the sememe-shared word group containing w^S_i. Then the loss function is:

$$\mathcal{L}_{sememe} = \sum_{w^S_i \in \mathbb{W}^S} \Big[\alpha_i \|\mathbf{w}^S_i - \hat{\mathbf{w}}^S_i\|^2 + \sum_{w^S_j \in \mathrm{Syn}(w^S_i)} \beta_{ij} \|\hat{\mathbf{w}}^S_i - \hat{\mathbf{w}}^S_j\|^2\Big], \tag{43}$$

where α and β are hyperparameters controlling the relative weights of the two terms. The second approach is named sememe embedding-based approach, which is inspired by the matrix factorization-based sememe prediction method SPSE. Different from SPSE, word embeddings are adjusted with sememe embeddings rather than fixed. Formally,

$$\mathcal{L}_{sememe} = \sum_{w^S_s \in \mathbb{W}^S, x_j \in \mathbb{X}} (\mathbf{w}^S_s \cdot \mathbf{x}_j + b_s + b'_j - \mathbf{M}^S_{sj})^2, \tag{44}$$

where $\mathbf{M}^S$ is the word-sememe annotation matrix of the source language, and b_s and b'_j are two learnable scalar parameter. In this way, the relations between words and sememes are injected into the word embeddings.

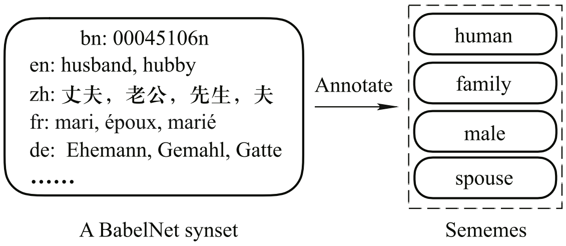


Fig. 7 Annotating sememes for a BabelNet synset. The synset comprises words in different languages expressing the same meaning "the man a woman is married to", and they share the four sememes on the right part

After obtaining the word embeddings of the source and target languages in a unified semantic space, cross-lingual sememe prediction can be conducted in a similar way to SPWE:

$$P(x_j|w_t^T) = \sum_{w_s^S \in \mathbb{W}^S} cos(\mathbf{w}_s^S, \mathbf{w}_t^T) \cdot \mathbf{M}_{sj}^S \cdot c^{r_s}. \tag{45}$$

3.2.2 Building a multilingual sememe KB based on BabelNet

BabelNet is a multilingual encyclopedic dictionary and comprises over 15 million *BabelNet synsets*. Each BabelNet synset is composed of words in multiple languages with the same meaning, i.e., multilingual synonyms, and they should have the same sememe annotation. Thus, building a multilingual sememe KB by annotating sememes for BabelNet synsets can actually provide sememe annotations for words in multiple languages simultaneously (Fig. 7 shows an example).

Qi et al. [30] build a seed sememe KB *BabelSememe* by manually annotating sememes for more than 15 thousand BabelNet synsets. They also propose the task of sememe prediction for BabelNet synsets (SPBS), aiming to automatically predict sememes for unannotated BabelNet synsets and gradually expand the seed KB.

They propose two models for SPBS, which utilize different information of synsets in BabelNet to predict sememes. The first model is SPBS with Semantic Representations (SPBS-SR). It is based on the semantic representations of synsets, which bears the semantic meanings of synsets. Specifically, they use the NASARI embeddings [45] provided by BabelNet, and adopts the similar prediction way to SPWE when given the target synset b:

$$P(x|b) = \sum_{b' \in \mathbb{B}'} \cos(\mathbf{b}, \mathbf{b}') \cdot \mathcal{I}_{\mathbb{X}(b')}(x) \cdot c^{r_{b'}}, \tag{46}$$

where $\mathbb{B}'$ is the set of synsets with known sememe annotations, and $\mathcal{I}_{\mathbb{X}(b')}(x)$ is a indicator function indicating whether a sememes x is annotated to the synset b'.

The second model uses the semantic relations between synsets in BabelNet and is named SPBS with Relational Representations (SPBS-RR). BabelNet contains many relations between synsets, and at the same time, HowNet includes relations between sememes. The relation between a pair of synsets is consistent with the relation between their respective sememes. Figure 8 gives an example. By introducing an extra relation "have_sememe" between synsets and their sememes, plus the synset-synset and sememe-sememe relations, all the synsets and sememes are connected together to form a semantic graph. Sememe prediction can be transformed to be a entity prediction task of knowledge graph, where the head entity (a target synset) and the relation ("have_sememe") are given and the tail entity are supposed to be predicted. They adopt a classical knowledge graph embedding method TransE [46] to learn the relational representations of synsets and sememes, and the training loss is:

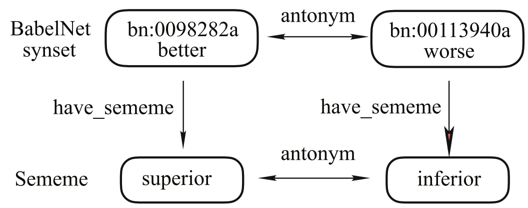


Fig. 8 An example of how relations between BabelNet synsets are consistent with the relations between respective sememes

$$\mathcal{L}_1 = \sum_{(h,r,t) \in \mathbb{G}} [\tau + d(\mathbf{h} + \mathbf{r}, \mathbf{t}) - d(\mathbf{h} + \mathbf{r}, \mathbf{t}')]_+, \tag{47}$$

where $[x]_+ = \max(0, x)$, scalar τ is a hyper-parameter, (h, r, t) is a triplet in the semantic graph $\mathbb{G}$ in which h, t can be a synset or sememe and r is a relation, (h, r, t') is a corrupted triplet, and $d(\mathbf{x}, \mathbf{y})$ is L_2 distance function:

$$d(\mathbf{x}, \mathbf{y}) = ||\mathbf{x} - \mathbf{y}||^2. \tag{48}$$

In addition, they introduce a special semantic equivalence relation between a synset and all its sememes, based on the fact that the meaning of a synset should be equal to the sum of its sememes' meanings. The corresponding loss is

$$\mathcal{L}_2 = \sum_{b' \in \mathbb{B}'} ||\mathbf{b}' + \mathbf{r}_s - \sum_{x \in \mathbb{X}(b')} \mathbf{x}||^2, \tag{49}$$

where r_s denotes the semantic equivalence relation. The overall training loss is as follows:

$$\mathcal{L} = \lambda_1 \mathcal{L}_1 + \lambda_2 \mathcal{L}_2, \tag{50}$$

where λ_1 and λ_2 are hyper-parameters controlling relative weights of the two losses. By optimizing the loss function, relational representations of all synsets, sememes and relations will be obtained. Then the sememe prediction score of a sememe x for a target synset b is

$$P(x|b) \propto d(\mathbf{b} + \mathbf{r}_h, \mathbf{x}), \tag{51}$$

where r_h is the "have_sememe" relation.

These two models SPBS-SR and SPBS-RR utilize different information and can be combined together to obtain better prediction results.

4 Conclusion and future directions

The importance of human knowledge to artificial intelligence, especially NLP, has been recognized by more and more studies. Sememe KBs have unique advantages in incorporation into deep neural networks and alleviating their bad performance in low-data regimes. Therefore, many studies on better utilizing sememes in deep learning-based NLP have been conducted in recent years and empirically demonstrated the effectiveness of sememes.

We believe that sememes will prove more powerful in future work. First, the structures of existing sememe annotations are rarely utilized, which capture more useful semantic information. It is challenging to incorporating structural sememes into neural networks, but it is also worthwhile. Second, the usefulness of sememes in low-data regimes has not been extensively exploited. The combination of sememes and few-shot learning or meta-learning is exciting and promising. Finally, the language universality of sememes is also worth exploring in some cross-lingual tasks.

As for expansion of sememe KBs, the work on building a multilingual sememe KB based on BabelNet is pioneering and will be seminal. It proposes a totally feasible and efficient way to annotate sememes for words and phrases in many languages. There are many research directions worthing exploration, e.g., using definitions to predict sememe for synsets and representation learning of synsets and sememes. In addition, the structures of sememes are also disregarded in previous sememe prediction work. It is a difficult but meaningful task to conduct structural sememe prediction.

There are some useful HowNet-related resources which are convenient for researches. *OpenHowNet* [47] is an open-source toolbox which provides some easy-to-use Python APIs of accessing data of HowNet. There is also a website of the same name supporting online search and display of sememes of words. Besides, we create a paper list named *SCPapers* which comprises all the must-read papers on sememe computation. Both OpenHowNet and SCPapers can be found on the GitHub page of THUNLP (Natural Language Processing Lab at Tsinghua University).

Acknowledgements This work was supported by the National Key Research and Development Program of China (2018YFB1004503) and the National Natural Science Foundation of China (NSFC Grant Nos. 61732008, 61532010). We also thank the anonymous reviewers for their comments.

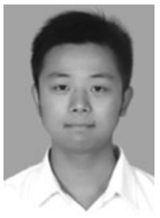
References

1. Bloomfield L. A set of postulates for the science of language. Language, 1926, 2(3): 153–164
2. Wierzbicka A. Semantics: Primes and Universals. Oxford: Oxford University Press, 1996
3. Dong Z, Dong Q. HowNet and the Computation of Meaning. Singapore: World Scientific Publishing, 2006
4. Gan K W, Wong P W. Annotating information structures in Chinese texts using HowNet. In: Proceedings of the 2nd Chinese Language Processing Workshop. 2000, 85–92
5. Liu Q, Li S. Word similarity computing based on HowNet. International Journal of Computational Linguistics & Chinese Language Processing, 2002, 7(2): 59–76
6. Zhang Y, Gong L, Wang Y. Chinese word sense disambiguation using HowNet. In: Proceedings of International Conference on Natural Computation. 2005, 925–932
7. Duan X, Zhao J, Xu B. Word sense disambiguation through sememe labeling. In: Proceedings of the 20th International Joint Conference on Artificial Intelligence. 2007, 1594–1599
8. Zhu Y, Min J, Zhou Y, Huang X, Wu L. Semantic orientation computing based on HowNet. Journal of Chinese Information Processing, 2006, 20(1): 14–20
9. Dang L, Zhang L. Method of discriminant for Chinese sentence sentiment orientation based on HowNet. Application Research of Computers, 2010, 4: 43
10. Fu X, Liu G, Guo Y, Wang Z. Multi-aspect sentiment analysis for Chinese online social reviews based on topic modeling and HowNet lexicon. Knowledge-Based Systems, 2013, 37: 186–195
11. Sun J, Cai D, Lv D, Dong Y. HowNet based Chinese question automatic classification. Journal of Chinese Information Processing, 2007, 21(1): 90–95
12. Moro A, Raganato A, Navigli R. Entity linking meets word sense disambiguation: a unified approach. Transactions of the Association for Computational Linguistics, 2014, 2: 231–244
13. Faruqui M, Dodge J, Jauhar S K, Dyer C, Hovy E, Smith N A. Retrofitting word vectors to semantic lexicons. In: Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. 2015, 1606–1615
14. Chen Q, Zhu X, Ling Z H, Inkpen D, Wei S. Neural natural language inference models enhanced with external knowledge. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics. 2018, 2406–2417
15. Sun M, Chen X. Embedding for words and word senses based on human annotated knowledge base: use HowNet as a case study. Journal of Chinese Information Processing, 2016, 30(6): 1–5
16. Niu Y, Xie R, Liu Z, Sun M. Improved word representation learning with sememes. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics. 2017, 2049–2058
17. Gu Y, Yan J, Zhu H, Liu Z, Xie R, Sun M, Lin F, Lin L. Language modeling with sparse product of sememe experts. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. 2018, 4642–4651
18. Zeng X, Yang C, Tu C, Liu Z, Sun M. Chinese LIWC lexicon expansion via hierarchical classification of word embeddings with sememe attention. In: Proceedings of the 32nd AAAI Conference on Artificial Intelligence. 2018, 5650–5657
19. Qi F, Huang J, Yang C, Liu Z, Chen X, Liu Q, Sun M. Modeling semantic compositionality with sememe knowledge. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. 2019, 5706–5715
20. Qin Y, Qi F, Ouyang S, Liu Z, Yang C, Wang Y, Liu Q, Sun M. Enhancing recurrent neural networks with sememes. 2019, arXiv preprint arXiv:1910.08910
21. Luo L, Ao X, Song Y, Li J, Yang X, He Q, Yu D. Unsupervised neural aspect extraction with sememes. In: Proceedings of the 28th International Joint Conference on Artificial Intelligence. 2019, 5123–5129
22. Zhang L, Qi F, Liu Z, Wang Y, Liu Q, Sun M. Multi-channel reverse dictionary model. In: Proceedings of the AAAI Conference on Artificial Intelligence. 2020, 312–319
23. Zang Y, Qi F, Yang C, Liu Z, Zhang M, Liu Q, Sun M. Word-level textual adversarial attacking as combinatorial optimization. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. 2020, 6066–6080
24. Xie R, Yuan X, Liu Z, Sun M. Lexical sememe prediction via word embeddings and matrix factorization. In: Proceedings of the 26th International Joint Conference on Artificial Intelligence. 2017, 4200–4206
25. Sarwar B, Karypis G, Konstan J, Riedl J. Item-based collaborative filtering recommendation algorithms. In: Proceedings of the 10th International Conference on World Wide Web. 2001, 285–295
26. Koren Y, Bell R, Volinsky C. Matrix factorization techniques for recommender systems. Computer, 2009, 42(8): 30–37
27. Jin H, Zhu H, Liu Z, Xie R, Sun M, Lin F, Lin L. Incorporating Chinese characters of words for lexical sememe prediction. In: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics. 2018, 2439–2449
28. Du J, Qi F, Sun M, Liu Z. Lexical sememe prediction by dictionary definitions and local semantic correspondence. Journal of Chinese Information Processing, 2020, 34(5): 1–9
29. Qi F, Lin Y, Sun M, Zhu H, Xie R, Liu Z. Cross-lingual lexical sememe

prediction. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing. 2018, 358–368

30. Qi F, Chang L, Sun M, Sicong O, Liu Z. Towards building a multilingual sememe knowledge base: predicting sememes for BabelNet synsets. In: Proceedings of the AAAI Conference on Artificial Intelligence. 2020, 8624–8631
31. Miller G A. WordNet: a lexical database for English. Communications of the ACM, 1995, 38(11): 39–41
32. Speer R, Chin J, Havasi C. Conceptnet 5.5: an open multilingual graph of general knowledge. In: Proceedings of the 31st AAAI Conference on Artificial Intelligence. 2017, 4444–4451
33. Mikolov T, Chen K, Corrado G, Dean J. Efficient estimation of word representations in vector space. In: Proceedings of 2013 International Conference on Learning Representations Workshop. 2013
34. Hochreiter S, Schmidhuber J. Long short-term memory. Neural Computation, 1997, 9(8): 1735–1780
35. Hinton G. Products of experts. In: Proceedings of the 9th International Conference on Artificial Neural Networks. 1999, 1–6
36. Pelletier F J. The principle of semantic compositionality. Topoi, 1994, 13(1): 11–24
37. Pelletier F J. Semantic compositionality. In: Oxford Research Encyclopedia of Linguistics. Oxford University Press, 2016
38. Mitchell J, Lapata M. Language models based on semantic composition. In: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing. 2009, 430–439
39. Socher R, Bauer J, Manning C D, Ng A Y. Parsing with compositional vector grammars. In: Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics. 2013, 455–465
40. Maas A L, Daly R E, Pham P T, Huang D, Ng A Y, Potts C. Learning word vectors for sentiment analysis. In: Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies. 2011, 142–150
41. Socher R, Perelygin A, Wu J Y, Chuang J, Manning C D, Ng A Y, Potts C. Recursive deep models for semantic compositionality over a sentiment treebank. In: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing. 2013, 1631–1642
42. Mitchell J, Lapata M. Vector-based models of semantic composition. In: Proceedings of ACL-08: HLT. 2008, 236–244
43. Navigli R, Ponzetto S P. BabelNet: the automatic construction, evaluation and application of a wide-coverage multilingual semantic network. Artificial Intelligence, 2012, 193: 217–250
44. Chen X, Xu L, Liu Z, Sun M, Luan H. Joint learning of character and word embeddings. In: Proceedings of the 24th International Joint Conference on Artificial Intelligence. 2015, 1236–1242
45. Camacho-Collados J, Pilehvar M T, Navigli R. Nasari: integrating explicit knowledge and corpus statistics for a multilingual representation of concepts and entities. Artificial Intelligence, 2016, 240: 36–64
46. Bordes A, Usunier N, Garcia-Duran A, Weston J, Yakhnenko O. Translating embeddings for modeling multi-relational data. In: Proceedings of the 26th Conference on Neural Information Processing Systems. 2013, 2787–2795
47. Qi F, Yang C, Liu Z, Dong Q, Sun M, Dong Z. OpenHowNet: an open sememe-based lexical knowledge base. 2019, arXiv preprint arXiv:1901.09957

Fanchao Qi is a PhD student of the Department of Computer Science and Technology, Tsinghua University, China. He got his BEng degree in 2017 from the Department of Electronic Engineering, Tsinghua University, China. His research interests include natural language processing and computational semantics. He has published papers in international conferences including AAAI, ACL and EMNLP.



Ruobing Xie is a researcher of WeChat, Tencent. He got his BEng degree in 2014 and his master degree in 2017 from the Department of Computer Science and Technology, Tsinghua University, China. His research interests are natural language processing and recommender system. He has published over 15 papers in international journals and conferences including IJCAI, AAAI, ACL and EMNLP.

Yuan Zang is an undergraduate student of the Department of Computer Science and Technology, Tsinghua University, China. His research interests lie in natural language processing and adversarial learning. He has published papers in ACL.

Zhiyuan Liu is an associate professor of the Department of Computer Science and Technology, Tsinghua University, China. He got his BEng degree in 2006 and his PhD in 2011 from the Department of Computer Science and Technology, Tsinghua University, China. His research interests are natural language processing and social computation. He has published over 40 papers in international journals and conferences including ACM Transactions, IJCAI, AAAI, ACL and EMNLP.

Maosong Sun is a professor of the Department of Computer Science and Technology, Tsinghua University, China. He got his BEng degree in 1986 and MEng degree in 1988 from Department of Computer Science and Technology, Tsinghua University, and got his PhD degree in 2004 from Department of Chinese, Translation, and Linguistics, City University of Hong Kong, China. His research interests include natural language processing, Chinese computing, Web intelligence, and computational social sciences. He has published over 150 papers in academic journals and international conferences in the above fields. He serves as the council member of China Computer Federation, the director of Massive Online Education Research Center of Tsinghua University, and the Editor-in-Chief of the Journal of Chinese Information Processing.